

Adobe Launch: Digioh JS Tag Installation Guide

Last Modified on 07/24/2026 10:59 am EDT

What You'll Learn in This Article

This guide explains how to install the Digioh JavaScript tag using Adobe Launch (Adobe Experience Platform Tags). The process mirrors the standard Google Tag Manager install: you create a single rule in your Launch property that loads the Digioh snippet sitewide. If you use Google Tag Manager instead, see [Google Tag Manager: Digioh JS Tag Installation Guide](#).

Before You Begin

You will need:

- Admin access to your Adobe Launch property tied to the domain where Digioh will run
 - Your Digioh JavaScript snippet, available in your Digioh account under **Account Settings > Install**
 - Access to publish to your Launch environments (Development, Stage, Production)
-

Step 1: Open Your Adobe Launch Property

1. Sign in to **Adobe Experience Platform Data Collection** at experience.adobe.com/data-collection.
2. Open the **Tags** section in the left navigation.
3. Click into the property tied to the domain where you want Digioh installed.

If you run multiple domains under separate properties, repeat the steps below for each property.

Step 2: Create a New Rule

1. In the property, go to the **Rules** tab.
 2. Click **Add Rule**.
 3. Name the rule **Digioh JS Tag**. A clear name makes future audits easier.
-

Step 3: Configure the Event

1. Under **Events**, click **Add**.
2. Set **Extension** to **Core**.
3. Set **Event Type** to **Library Loaded (Page Top)**.

4. Click **Keep Changes**.

The **Library Loaded (Page Top)** event fires as early as possible in the page lifecycle. This lets Digioh render above-the-fold experiences without flicker.

Step 4: Add the Digioh Snippet as a Custom Code Action

1. Under **Actions**, click **Add**.
2. Set **Extension** to **Core**.
3. Set **Action Type** to **Custom Code**.
4. Set the **Language** to **JavaScript**.
5. Click **Open Editor**.
6. Paste your Digioh snippet from **Account Settings > Install** in your Digioh account.
7. Save and close the editor.
8. Click **Keep Changes**.

Leave **Execute Globally** unchecked. The default behavior is correct. **Optional:** If you want Digioh to skip certain paths (for example, your checkout flow), add a **Path** condition under **Conditions** in the rule before saving.

Step 5: Save and Publish

1. Click **Save** to save the rule.
 2. Open the **Publishing Flow** tab.
 3. Add your rule to a new or existing **Library**.
 4. Build the library to your **Development** environment for QA.
 5. Once verified, submit for approval and publish through **Stage** and **Production** using your team's normal release process.
-

Step 6: Verify the Installation

After publishing, load a page on your published environment and open your browser developer tools. In the **Console** tab, type: DIGIOH_API If the console returns an object, your Digioh tag is loaded. You can also install the [Adobe Experience Platform Debugger](#) Chrome extension and

confirm that the **Digioh JS Tag** rule appears in the executed rules list for the page. In your Digioh dashboard, open any active campaign and check the analytics. Impressions will begin appearing within a few minutes of the first page view. Your Digioh JavaScript tag is now live!

Important: Adobe Launch and Ad Blockers

Some ad blockers will block scripts loaded through Adobe Launch, which can prevent Digioh from running for those visitors. If you see lower-than-expected impressions, this is the most common cause. For sites that need maximum coverage, consider also installing the Digioh tag directly in your site template alongside the Launch install.

Important: Ordering with Other Adobe Tags

If you also use Adobe Analytics, Adobe Target, or the Adobe Experience Platform Web SDK in the same Launch property, the default ordering works in most cases. Digioh runs independently of Adobe's personalization and analytics calls. If you plan to use Adobe Target audiences or Adobe Analytics segments as targeting conditions inside Digioh, expose those values in your data layer or in a first-party cookie. You can then use them as key/value conditions on your Digioh campaigns.

Troubleshooting

DIGIOH_API is undefined after publishing. Confirm the rule is in the library that was published to the environment you are testing. The Launch debugger will show which libraries loaded on the current page. **Campaigns are not appearing on the site.** Verify the campaign is published in Digioh and that its display conditions match the page you are testing. The most common cause is a path or device condition excluding the test URL. **Digioh loads but renders late.** Confirm the rule event is set to **Library Loaded (Page Top)**. The **DOM Ready** and **Window Loaded** events both fire later in the page lifecycle and can cause visible flicker. **The Custom Code editor is rejecting the snippet.** Make sure the language is set to **JavaScript**, not HTML. The Digioh snippet is pure JavaScript and does not include `<script>` tags inside the editor.
