



## Getting Started

What is Segment?  
[How Segment Works](#)  
Getting Started Guide  
A Basic Segment Installation  
Planning a Full Installation  
A Full Segment Installation  
Sending Data to Destinations  
Testing and Debugging  
What's Next  
Use Cases

## Guides

## Connections

## Unify

## Engage

## Privacy

## Protocols

## Segment App

## API

## Partners

## Glossary

## Config API

## Help

or feature announcements in minutes instead of weeks. The Appcues JavaScript Destination is open-source. You can browse the code [on GitHub](#).

## Getting Started

1. From the Segment web app, click **Catalog**.
2. Search for "Appcues" in the Catalog, select it, and choose the source you'll connect to the destination.
3. In the destination settings, enter your Account ID (for client-side integration functionality) and/or your API Key (for server-side integration functionality) from the [Appcues account page](#).

## Server

As an alternative to a traditional JavaScript implementation, Appcues offers a server-side destination with Segment.

You may find the server-side destination useful if you'd like to send user profile or event data to Appcues from another Segment partner service. You can use the server-side destination alongside the JavaScript destination, which you may find preferable to routing all data through the JavaScript destination.

As with the JavaScript destination, you can segment and target user profile and event data received through the Appcues server-side Segment destination.

For example, using the server-side destination, you can direct customer profile and event data from a CRM tool into Appcues. You can then use the directed data for content targeting and user segmentation in the Appcues content editor, alongside data from Segment's `analytics.js` destination.



#### Implementations can only be used with server or mobile sources

Analytics.js sources will default to a device-mode connection.

## Page

Refer to the Segment Spec for information about the [Page method](#). The following represents an example `page` call:

```
analytics.page();
```

Each time the page changes, Appcues checks to see if a user qualifies for an experience. When you first call `page` using `Analytics.js`, `Appcues.start` checks if there are any current flows for the user and loads them, if necessary.

## Identify

Refer to the Segment Spec for information about the [Identify method](#). The following represents an example `identify` call:

```
analytics.identify('12091906-01011992', {
  name: 'John Joe',
  email: 'john.doe@example.com'
});
```

When you call `identify` with `Analytics.js`, Segment calls `Appcues.identify`. This is the preferred method of using and targeting on user properties.

To get the most out of Appcues, you should send as much user data as possible in the `identify` call. Properties personalize content and target experiences to specific users. Most Appcues customers send properties that fall into one of the following groups:

- Properties to target based on broad classifications, such as `role` or `userType`
- Properties to personalize Appcues content, such as `name`, `firstName`, or `company`
- Properties to target based on user lifecycle, such as `createdAt` (date), or usage metrics, such as `numTasksComplete`

## Group

Refer to the Segment Spec for information about the [Group method](#). The following represents an example `group` call:

```
analytics.group("0e8c78ea9d97a7b8185e8632", {
  name: "Initech",
  industry: "Technology",
  employees: 329,
  plan: "enterprise",
  "total billed": 830
});
```

When you call `group` with `Analytics.js`, Segment calls `Appcues.group`. This is the preferred method of using and targeting on account-level properties.

## Track

Refer to the Segment Spec for information about the [Track method](#). The following represents an example `track` call:

```
analytics.track("step_activated", {  
  property: "test"  
});
```

Calls to `analytics.track` also invoke `Appcues.track`. As a result, the call sends event data to Appcues, where you can use it for future content triggering.

## Sending Appcues events to other Segment partner services

Appcues supports sending events to other tools on the Segment platform. This capability allows you to read Appcues events in your third-party analytics or marketing automation tools.

These events are sent as track calls to the other destinations you've turned on. The following shows a partial list of trackable Appcues content lifecycle events:

- `flow_shown`
- `flow_skipped`
- `flow_finished`
- `flow_form_submission`
- `form_field_submission`
- `step_activated`
- `hotspots_shown`
- `hotspots_skipped`
- `hotspots_completed`
- `hotspot_activated`
- `coachmarks_shown`
- `coachmarks_completed`

To enable this feature, go to the Integrations Settings in Appcues and click "Activate" under the Segment integration.

After you activate the Segment integration, Appcues forwards these events to your Segment workspace. The events will appear in the same [Source Debugger](#).

## Whitelisted Domains

By default, Appcues targets based on the URL's path.

For example, if you create an Appcues experience and target it to `/integrations`, it appears wherever the embed script is installed with that URL path, like `appcues.com/integrations` and `my.appcues.com/integrations`.

If you've installed the `Analytics.js` script on multiple domains (like a marketing site and your web app), you should use Appcues-whitelisted domains when targeting your experience.

## Delete

When you trigger a user deletion using [Segment's Privacy features](#), Segment forwards an actionable delete

notification to Appcues. Refer to the Appcues documentation for information on [how Appcues handles deletion requests](#).

## Engage

You can send computed traits and audiences generated using [Engage](#) to this destination as a **user property**. To learn more about Engage, schedule a [demo](#).

For user-property destinations, an [identify](#) call is sent to the destination for each user being added and removed. The property name is the snake\_cased version of the audience name, with a true/false value to indicate membership. For example, when a user first completes an order in the last 30 days, Engage sends an Identify call with the property `order_completed_last_30days: true`. When the user no longer satisfies this condition (for example, it's been more than 30 days since their last order), Engage sets that value to `false`.

When you first create an audience, Engage sends an Identify call for every user in that audience. Later audience syncs only send updates for users whose membership has changed since the last sync.



### Real-time to batch destination sync frequency

Real-time audience syncs to Appcues may take six or more hours for the initial sync to complete. Upon completion, a sync frequency of two to three hours is expected.

## Settings

Segment lets you change these destination settings from the Segment app without having to touch any code.

SETTING	DESCRIPTION
API Key <i>(required)</i>	<code>string</code> · <b>Required for server-side integration functionality.</b> You can find your API Key in your <a href="#">Appcues account page</a> .
Appcues Id <i>(required)</i>	<code>string</code> · <b>Required for client-side integration functionality.</b> You can find your Appcues ID in your <a href="#">Appcues account page</a> .

This page was last modified: 23 Oct 2023

## Need support?

Questions? Problems? Need more info? Contact Segment Support for assistance!

[Visit our Support page](#)

## Help improve these docs!

[Edit this page](#)

[Request docs change](#)

## Was this page helpful?

☒ Yes

☐ No

# Get started with Segment

Segment is the easiest way to integrate your websites & mobile apps data to over 300 analytics and growth tools.

Your work e-mail

Request Demo

or

Create free account

© 2025 Segment.io, Inc.

[Privacy](#)

[Terms](#)

[Website Data Collection Preferences](#)

