helps engineering and product teams ensure that new features will improve the customer experience without breaking or degrading performance and/or business metrics.

This destination is maintained by A/B Smartly. For any issues with the destination, contact A/B Smartly's Support.

Implementation Prerequisite

A/B Smartly works differently than other Segment destinations. Because A/B Smartly SDKs are used to modify and deliver experiences to users, you must implement them at a point in your website or app that allows them to make visual modifications for users.

A/B Smartly requires you to integrate the A/B Smartly SDKs natively, before the Segment snippet or implementation.

Even though Segment maps `track`, and in some cases `page` and `screen` events to A/B Smartly's `track` events, you must implement the snippet on your site to ensure that experiments run.

Segment provides specific implementation details for A/B Smartly in the sections below, in addition to details of

the out-of-the-box mappings that Segment's A/B Smartly integration handles transparently.

Getting Started

1. From the Destinations catalog page in the Segment App, click **Add Destination**.

2. Search for "A/B Smartly" in the Destinations Catalog, and select the "A/B Smartly" destination.

3. Choose which Source should send data to the "A/B Smartly" destination.

4. Go to the A/B Smartly dashboard(https://your-org-name.absmartly.com/apikey/list){:target="_blank"}, find and copy the "API key" that you created for segment.

5. Enter the "API Key" in the "A/B Smartly" destination settings in Segment.

6. the integration requests for an Application name go to your A/B Smartly dashboard (`https://your-org-name.absmartly.com/application/create`) and create an Application named "Segment", or whatever you would like to call it. Use that name in the Application field of the integration settings.

7. Add also your A/B Smartly Collector endpoint. It's the same endpoint that you are using in all your A/B Smartly SDKs.

8. Enter the environment from your Environment list (`https://your-org-name.absmartly.com/environment/list`) that should receive your Segment data. Most likely it will be the production one.

9. And finally a mapping of Segment Identities to A/B Smartly Units(`https://your-org-name.absmartly.com/unit/list`). You should map all of your Segment identities that you would like to use in your A/B tests to the units that you already have in A/B Smartly. Users must map all the identity types, but not for the individual users. Map all of your ids in your Unit List(`https://your-org-name.absmartly.com/unit/list`).

10. Optionally go to "Goal Mapping" and start adding the track calls that you would like to see showing up on A/B Smartly as goals. You only need to create a name mapping if the name of the goal on A/B Smartly's platform is different from the name of the track call in Segment.

Page

Take a look at the Page method documentation to understand what it does. An example call would look like:

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

Segment sends Page calls to A/B Smartly as a `Page` goal. The goal name is `Page: <Page Name>`. For example, `Page: Home`. If the goal doesn't exist in the A/B Smartly web console, it is ignored.

Screen

Take a look at the Screen method documentation to understand what it does. An example call would look like:

```
[[SEGAnalytics sharedAnalytics] screen:@"Home"];
```

Segment sends Screen calls to A/B Smartly as a `Screen` goal. The goal name is `Screen: <Screen Name>`. For example, `Screen: Login`. If the goal doesn't exist in the A/B Smartly web console, it is ignored.

Track

Take a look at the Track method documentation to understand what it does. An example call would look like:

```
analytics.track("Login Button Clicked");
```

Segment sends Track calls to A/B Smartly as a `track` event. A/B Smartly's track calls are the way to track goals. `analytics.track('booking')` is equivalent to an A/B Smartly SDK track call `context.track('booking')`.

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.

> ℹ️ **Real-time to batch destination sync frequency**
>
> Real-time audience syncs to AB Smartly 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 *(required)*	`string`. Find and create API keys in the settings page of the A/B Smartly web console.
Collector Endpoint *(required)*	`string`. Endpoint of the A/B Smartly Collector. Example: https://your-company.absmartly.iop/v1
Enable App Screen View Tracking	`boolean`, defaults to `FALSE`. Send Segment Screen view events to A/B Smartly as goals. The goal name will be `<screen_name>`_screenview. The screen name is lower-cased and any spaces or special characters are replaced with underscores. For example, a view of the "Home" screen will trigger the home_screenview goal. If the goal does not exist in the A/B Smartly web console, it is ignored.
Enable Exposure Tracking	`boolean`, defaults to `FALSE`. Send Group event payloads directly to the A/B Smartly collector.
Enable Page View Tracking	`boolean`, defaults to `FALSE`. Send Segment Page view events to A/B Smartly as goals. The goal name will be `<page_name>`_pageview. The page name is lower-cased and any spaces or special characters are replaced with underscores. For example, a view of the "Home" page will trigger the home_pageview goal. If the goal does not exist in the A/B Smartly web console, it is ignored.
Environment *(required)*	`string`. Find and create environments in the settings page of the A/B Smartly web console.
Goal Mapping	`text-map`, defaults to {}. Optional mapping of Segment event to A/B Smartly goal. Find and create goals in the settings page of the A/B Smartly web console.
Unit Mapping *(required)*	`text-map`, defaults to {}. Mapping of Segment identities to A/B Smartly units. Find and create unit types in the settings page of the A/B Smartly web console.

This page was last modified: 30 Aug 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