



Documentation

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

Partner Owned

- This integration is partner owned. Please reach out to the partner's support for any issues.

[Bloomreach Engagement](#) is a Customer Data & Experience Platform (CDXP) which creates a unified source of customer intelligence in real-time, ready for immediate activation using its own built-in omnichannel marketing systems (web, email, push, mobile, text messages, etc.) powered by customer-centric analytics and artificial intelligence (product recommendations and predictions).

This destination is maintained by Bloomreach Engagement. For any issues with the destination, contact [the Bloomreach Engagement Support team](#).

Getting Started

1. From the Segment web app, click **Catalog**.

2. Search for "Bloomreach Engagement" in the Catalog, select it, and choose which of your sources to connect the destination to.

3. Create a [public API group](#) for your Segment integration in your Bloomreach Engagement project. Don't forget

to set the appropriate [group permissions](#) so you can receive events and customer updates.

4 Fill in the "API Base URL", "API key" and "Project Token" into your Segment Settings UI. You can find all of the above in the API settings page of your Bloomreach Engagement project.

5 Enter your Bloomreach Engagement hard ID and soft ID names into the corresponding fields to specify Segment's `userId` and `anonymousId` mapping into your Bloomreach Engagement ID structure.

Common fields

If you have not had a chance to review the Segment spec, take a look to understand what the [Common fields](#) are.

The `userId` and `anonymousId` common fields are used for all types of calls to identify the user in Bloomreach Engagement. Mapping of the IDs is based on the destination settings:

SEGMENT	BLOOMREACH ENGAGEMENT
<code>userId</code>	Bloomreach Engagement hard ID (e.g registered)
<code>anonymousId</code>	Bloomreach Engagement soft ID (e.g cookie)

Other common fields are used only for `track`, `page` and `screen` calls which are translated into Bloomreach Engagement events. The following common fields are mapped to Bloomreach Engagement:

SEGMENT	BLOOMREACH ENGAGEMENT
<code>timestamp</code>	<code>timestamp</code> (string date is parsed to unix timestamp)
<code>context: app, device, os, screen, referrer, campaign, user_agent, location</code>	<code>event properties</code> (fields that contain child objects are flattened)

Page

If you have not had a chance to review the Segment spec, take a look to understand what the [Page method](#) does.

Page calls will be sent to Bloomreach Engagement as a `page_visit` event with the `properties` field mapped into event properties and the `name` field mapped into the `page_name` property.

Example of page call:

```
analytics.page("Home", {
  url: "https://Bloomreach Engagement.com",
  referrer: "http://google.com"
})
```

This `page` call is translated into a `page_visit` event with the following properties:

```
"page_name": "Home",
"url": "https://Bloomreach Engagement.com",
"referrer": "http://google.com"
```

An optional event `session_ping` can be tracked along with `page_visit` for [automatic session tracking](#). You can adjust this behavior in your Bloomreach Engagement destination settings by toggling on and off the 'Track session ping' settings. The Bloomreach Engagement soft ID must be set to `cookie` and the `anonymousId` field must be present in the `page` call for session events to work.

Screen

If you have not had a chance to review the Segment spec, take a look to understand what the [Screen method](#)

does.

Screen calls will be sent to Bloomreach Engagement as a `screen_visit` event with the `properties` field mapped into event properties and the `name` field mapped into the `screen_name` property.

Example of screen call:

```
[[SEGAnalytics sharedAnalytics] screen:@"Home"  
  properties:@{ @"Feed Type": @"private" }];
```

This screen call is translated into a `screen_visit` event with the following properties:

```
"screen_name": "Home",  
"Feed Type": "private"
```

An optional event `session_ping` can be tracked along with `screen_visit` for [automatic session tracking](#). You can adjust this behavior in your Bloomreach Engagement destination settings by toggling on and off the 'Track session ping' settings. The Bloomreach Engagement soft ID must be set to `cookie` for session events to work and `anonymousId` field must be present in the screen call for session events to work.

Track

If you have not had a chance to review the Segment spec, take a look to understand what the [Track method](#) does.

Track calls will be sent to Bloomreach Engagement as events under name provided in the event field. The `properties` field will be mapped into event properties (objects will be flattened using underscore).

Example of track call:

```
analytics.track("Registered", {  
  plan: "Pro Annual",  
  accountType: "Facebook"  
});
```

This track call is translated into a `Registered` event with the following properties:

```
"plan": "Pro Annual",  
"accountType": "Facebook"
```

Identify

If you have not had a chance to review the Segment spec, take a look to understand what the [Identify method](#) does.

Identify calls will be sent to Bloomreach Engagement as customer updates with traits set as customer properties.

Example of identify call:

```
analytics.identify("userId123", {  
  name: "John Doe",  
  email: "john.doe@example.com",  
  address: {  
    city: "New York",  
    country: "USA"  
  }  
});
```

This identify call is translated into a customer update for user with Bloomreach Engagement hard id `userId123` with properties:

```
"name": "John Doe",
"email": "john.doe@example.com",
"address_city": "New York",
"address_country": "USA",
```

Alias

If you have not had a chance to review the Segment spec, take a look to understand what the [Alias method](#) does.

The alias call can be used to merge two user identities and their data to one. The `previousId` field should always contain a previously used `anonymousId`, as merging users by specifying two `userIds` is not supported. Sending an alias event with `previousId` and no `userId` will cause the event to be ignored. Note that users are also merged when any call specifies both a `userId` and an `anonymousId`, which previously belonged to two separate users.

Example of alias call:

```
analytics.alias("507f191e81");
```

Group

If you have not had a chance to review the Segment spec, take a look to understand what the [Group method](#) does.

Group calls will be sent to Bloomreach Engagement as customer updates with group traits as customer properties prefixed with `group_` and `groupId` into `group_id`. For example:

```
analytics.group("123", {
  name: "Bloomreach Engagement",
  industry: "Technology"
});
```

will be translated into a customer update with properties:

```
"group_id": "123",
"group_name": "Bloomreach Engagement",
"group_industry": "Technology",
```

Disclaimer: This is a beta version of group tracking and might be subject to change.

General

Nested Objects

Values that contain nested objects will be flattened using underscore.

For example:

```
analytics.identify('userId123', {
  address: {
    city: "New York",
    country: "USA"
  }
});
```

The properties would be sent as:

```
"address_city": "New York",
"address_country": "USA",
```

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 Bloomreach Engagement 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 Base URL <i>(required)</i>	<code>string</code> . Exponea endpoint URL (default <code>https://api.exponea.com/</code>)
API Key <i>(required)</i>	<code>string</code> . Public key (find more here https://docs.exponea.com/reference#section-setting-up-public-key-in-exponea-app)
Exponea hard ID <i>(required)</i>	<code>string</code> . Specify hard id which will be used in Exponea, typical name is 'registered'. The id must be already created in Exponea project.
Exponea soft ID <i>(required)</i>	<code>string</code> . Specify soft id which will be used in Exponea, typical name is 'cookie'. The id must be already created in Exponea project.
Flatten nested objects	<code>boolean</code> , defaults to <code>FALSE</code> . Turn this setting on if you want to apply object flattening on customer traits and event properties.
Project token <i>(required)</i>	<code>string</code> . Exponea project token
Track session ping	<code>boolean</code> , defaults to <code>FALSE</code> . Track an additional <code>session_ping</code> event with each 'page' and 'screen' events. This will enable automatic <code>session_start</code> and <code>session_end</code> tracking in Exponea. The Exponea soft ID must be set to 'cookie' for session events to work.

This page was last modified: 11 Jul 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](#)

