



## 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

#### Additional versions of this destination are available

This page is about the Mixpanel (Actions) Destination. See below for information about other versions of the Mixpanel destination:

[Mixpanel \(Classic\)](#)

[Mixpanel](#) enables you to build better products through powerful, self-serve product analytics to help you convert, engage, and retain more users.

## Benefits of Mixpanel (Actions) vs Mixpanel Classic

Mixpanel (Actions) provides the following benefits over the classic Mixpanel destination:

- **Simpler setup** - Mixpanel (Actions) has a streamlined default setup process making it easier to get started in a way that "just works".
- **More control** - Actions-based destinations enable you to define the mapping between the data Segment receives from your source, and the data Segment sends to the destination.
- **Additional default property mappings** - More default mappings from the Segment context like app name,

app namespace, device type, and more.

**Improved Groups support** - Implementation of [Segment Groups](#) with [Mixpanel Group Analytics](#) is easier. If you're already using Segment Groups, no code changes are required.

**E-commerce mappings** - Mixpanel (Actions) accepts products nested within arrays in the `Order Completed` event as described in the Segment [ecommerce spec](#).

**Batching Requests** - If you have a lot of events, Mixpanel (Actions) provides more efficient way to receive and process those large sets of data.

## Getting started

1. Go to your [Mixpanel project settings](#). Copy the Mixpanel API Key and API Secret for your project.

2. From the Segment web app, click **Catalog**, then click **Destinations**.

3. Find the Destinations Actions item in the left navigation, and click it.

4. Click the "Mixpanel" item to select it and click **Configure**.

5. Choose which of your sources to connect the destination to. (You can connect more sources to the destination later.)

## Connection Modes for Mixpanel (Actions) destination

The Mixpanel (Actions) destination does not offer a device-mode connection mode. If you're using one of Segment's new libraries ([Analytics.js 2.0](#), [Swift](#) or [Kotlin](#)) with the Actions-framework version of the destination, you do not need the device-mode connection.

## Destination Settings

SETTING	DESCRIPTION
Data Residency	Learn about <a href="#">EU data residency</a> and <a href="#">India data residency</a>
Secret Key	<i>Required.</i> Mixpanel project secret.
Project Token	<i>Required.</i> Mixpanel project token.
Source Name	This value, if it's not blank, will be sent as <code>segment_source_name</code> to Mixpanel for every event/page/screen call.
Strict Mode	This value, if it's 1 (recommended), Mixpanel will validate the events you are trying to send and return errors per event that failed. Learn more about the Mixpanel <a href="#">Import Events API</a>

## Available Presets

Mixpanel (Actions) has the following presets:

PRESET NAME	TRIGGER	DEFAULT ACTION
Page Calls	Event type = "page"	Track Event
Track Calls	Event type = "track" and event != "Order Completed"	Track Event
Identify Calls	Event type = "identify"	Identify User
Group Calls	Event type = "group"	Group Identify User

PRESET NAME	TRIGGER	DEFAULT ACTION
Order Completed Calls	Event type = "track" and event = "Order Completed"	Track Purchase
Screen Calls	Event type = "screen"	Track Event

## Available Actions

Build your own Mappings. Combine supported [triggers](#) with the following Mixpanel-supported actions:

*i*

**Mapping limits per destination**  
Individual destination instances have support a maximum of 50 mappings.

- Alias
- Group Identify User
- Track Event
- Identify User
- Track Purchase
- Increment Properties

### Alias

Create an alias to a distinct id. This action is primarily supported for the sake of customers using the legacy identity management in their Mixpanel project. For new customers or those who have [migrated](#) to the [new identity management](#) in Mixpanel should use `identify`.

Alias is a **Cloud** action.

Click to show / hide fields

FIELD	DESCRIPTION
Alias	Type: <code>STRING</code> A new distinct id to be merged with the original distinct id. Each alias can only map to one distinct id.
Distinct ID	Type: <code>STRING</code> A distinct id to be merged with the alias.

### Group Identify User

Updates or adds properties to a group profile. The profile is created if it does not exist. [Learn more about Group Analytics](#).

Group Identify User is a **Cloud** action. The default Trigger is: `type = "group"`

Click to show / hide fields

FIELD	DESCRIPTION
Group Key	Type: <code>STRING</code> The group key you specified in Mixpanel under Project settings. If this is not specified, it will be defaulted to <code>"\$group_id"</code> .
Group ID *	Type: <code>STRING</code> The unique identifier of the group. If there is a trait that matches the group key, it will override this value.



to set a trait where the key of the trait is the group key and the value of that trait is the group id. While that is no longer necessary, this behavior is supported in the Mixpanel (Actions) destination to ensure backwards-compatibility. *If* you specify a trait that matches specified group key, Mixpanel uses the value of that trait as the group id.

Scenario 1: No group key trait specified

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

The group id that Mixpanel will use is 0e8c78ea9d97a7b8185e8632.

Scenario 2: Group key trait IS specified

```
analytics.group("0e8c78ea9d97a7b8185e8632", {
  $group_id: "12345",
  name: "Initech",
  industry: "Technology",
  employees: 329,
  plan: "enterprise",
  "total billed": 830
});
```

The group id that Mixpanel will use is 12345.

The below special traits will be mapped to Mixpanel reserved properties automatically to fit Mixpanel’s use cases.

traits.name -> \$name.

Track Event

Send an event to Mixpanel. [Learn more about Events in Mixpanel](#)

Track Event is a **Cloud** action. The default Trigger is: type = "track"

Click to show / hide fields

FIELD	DESCRIPTION
Event Name *	Type: STRING The name of the action being performed.
Distinct ID	Type: STRING A distinct ID specified by you.
Anonymous ID	Type: STRING A distinct ID randomly generated prior to calling identify.
User ID	Type: STRING The distinct ID after calling identify.
Group ID	Type: STRING The unique identifier of the group that performed this event.
Insert ID	Type: STRING A random id that is unique to an event. Mixpanel uses \$insert_id to deduplicate events.
Timestamp	Type: DATETIME The timestamp of the event. Mixpanel expects epoch timestamp in millisecond or second. Please note, Mixpanel only accepts this field as the timestamp. If the field is empty, it will be set to the time Mixpanel servers receive it.

FIELD	DESCRIPTION
App Name	Type: <code>STRING</code> The name of your application.
App Namespace	Type: <code>STRING</code> The namespace of your application.
App Build	Type: <code>STRING</code> The current build of your application.
App Version	Type: <code>STRING</code> The current version of your application.
OS Name	Type: <code>STRING</code> The name of the mobile operating system or browser that the user is using.
OS Version	Type: <code>STRING</code> The version of the mobile operating system or browser the user is using.
Device ID	Type: <code>STRING</code> A unique identifier for the device the user is using.
Device Type	Type: <code>STRING</code> The type of the user's device.
Device Name	Type: <code>STRING</code> The name of the user's device.
Device Manufacturer	Type: <code>STRING</code> The device manufacturer that the user is using.
Device Model	Type: <code>STRING</code> The device model that the user is using.
Bluetooth Enabled	Type: <code>BOOLEAN</code> Whether bluetooth is enabled.
Carrier	Type: <code>STRING</code> The carrier that the user is using.
Cellular Enabled	Type: <code>BOOLEAN</code> Whether cellular is enabled.
...	Type: <code>BOOLEAN</code>

## Identify User

Set the user ID for a particular device ID or update user properties. Learn more about [User Profiles](#) and [Identity Management](#).

Identify User is a **Cloud** action. The default Trigger is: `type = "identify"`

Click to show / hide fields

FIELD	DESCRIPTION
IP Address	Type: <code>STRING</code> The IP address of the user. This is only used for geolocation and won't be stored.
User ID	Type: <code>STRING</code> The unique user identifier set by you
Anonymous ID	Type: <code>STRING</code> The generated anonymous ID for the user

FIELD	DESCRIPTION
User Properties	Type: <code>OBJECT</code> Properties to set on the user profile



Segment maps the `userId` set in the `identify` event to the `distinct ID` in Mixpanel. Segment also maps the following traits to Mixpanel reserved properties to fit Mixpanel's use cases: `traits.created` -> `$created`, `traits.email` -> `$email`, `traits.firstName` -> `$first_name`, `traits.lastName` -> `$last_name`, `traits.name` -> `$name`, `traits.username` -> `$username` and `traits.phone` -> `$phone`.

## Track Purchase

Send an 'Order Completed' Event to Mixpanel.

Track Purchase is a **Cloud** action. The default Trigger is: `type = "track"`

Click to show / hide fields

FIELD	DESCRIPTION
Generate Purchase Event Per Product	Type: <code>BOOLEAN</code> When enabled, send "Product Purchased" with each product within the event.
Distinct ID	Type: <code>STRING</code> A distinct ID specified by you.
Anonymous ID	Type: <code>STRING</code> A distinct ID randomly generated prior to calling <code>identify</code> .
User ID	Type: <code>STRING</code> The distinct ID after calling <code>identify</code> .
Group ID	Type: <code>STRING</code> The unique identifier of the group that performed this event.
Insert ID	Type: <code>STRING</code> A random id that is unique to an event. Mixpanel uses <code>\$insert_id</code> to deduplicate events.
Timestamp	Type: <code>DATETIME</code> The timestamp of the event. Mixpanel expects epoch timestamp in millisecond or second. Please note, Mixpanel only accepts this field as the timestamp. If the field is empty, it will be set to the time Mixpanel servers receive it.
App Name	Type: <code>STRING</code> The name of your application.
App Namespace	Type: <code>STRING</code> The namespace of your application.
App Build	Type: <code>STRING</code> The current build of your application.
App Version	Type: <code>STRING</code> The current version of your application.
OS Name	Type: <code>STRING</code> The name of the mobile operating system or browser that the user is using.
OS Version	Type: <code>STRING</code> The version of the mobile operating system or browser the user is using.
Device ID	Type: <code>STRING</code> A unique identifier for the device the user is using.

FIELD	DESCRIPTION
Device Type	Type: <code>STRING</code> The type of the user's device.
Device Name	Type: <code>STRING</code> The name of the user's device.
Device Manufacturer	Type: <code>STRING</code> The device manufacturer that the user is using.
Device Model	Type: <code>STRING</code> The device model that the user is using.
Bluetooth Enabled	Type: <code>BOOLEAN</code> Whether bluetooth is enabled.
Carrier	Type: <code>STRING</code> The carrier that the user is using.
Cellular Enabled	Type: <code>BOOLEAN</code> Whether cellular is enabled.

When set `Generate Purchase Event Per Product` to `true`, this setting effectively “flattens” the array of objects in the `Order Completed`'s `products` property by tracking a `Product Purchased` event for each item in the array. This enables more sophisticated analysis on a per-product basis in Mixpanel. These `Product Purchased` events will contain all of the key-value pairs from their respective object in the `products` array as event properties, along with the `order_id` and `checkout_id` from the `Order Completed` event.

### Increment Properties

Increment the value of a user profile property. [Learn More](#).

Increment Properties is a **Cloud** action. The default Trigger is: `type = "track"`

Click to show / hide fields

FIELD	DESCRIPTION
IP Address	Type: <code>STRING</code> The IP address of the user. This is only used for geolocation and won't be stored.
User ID	Type: <code>STRING</code> The unique user identifier set by you
Anonymous ID	Type: <code>STRING</code> The generated anonymous ID for the user
Increment Numerical Properties *	Type: <code>OBJECT</code> Object of properties and the values to increment or decrement. For example: <code>{“purchases”: 1, “items”: 6}</code> .

**Anonymous ID format**  
Mixpanel requires that values it receives for the anonymous identifier (`anonymousId` in Segment) must be in the UUID v4 format. Analytics.js sends `anonymousId` in this format by default. If you manually send anonymous identifiers to Mixpanel, ensure they are in the correct format.


### Migration from Mixpanel Classic

Assuming you're already using Segment Cloud-mode, the Mixpanel (Actions) destination is expected to have no breaking changes when upgrading. With the exception of a few new properties added to your events in the new



Actions destination, there should be no difference in the data received in Mixpanel when using either of the Mixpanel destinations.

If you want to confirm, you can configure the new destination to point to a different Mixpanel project and connect it to the same source(s) as the Classic destination and manually verify before fully switching over.



Contact Mixpanel support if you find features missing from the Mixpanel (Actions) destination that were available in the classic Mixpanel destination.

## Mixpanel settings mapping

Search for setting..

All	Cloud	Device-web	Device-mobile
MIXPANEL CLASSIC DESTINATION SETTING		HOW TO ENABLE IN MIXPANEL (ACTIONS)	
CONNECTION SETTINGS			
EVENT TRACKING			
TRAITS & PROPERTIES			
IDENTITY RESOLUTION			
OTHER SETTINGS			

## Troubleshooting

### Track events are not attributed to Mixpanel Groups

If the Mixpanel (Actions) destination uses \$group\_id as the group key, ensure that the mappings handling your track events have the field for **Group ID** mapped to a valid value. By default, this field maps to the event variable context.groupId.

To send Track events with a custom Group Key, include the key as a property of Track events. For example:

```
analytics.track('Example Event', { custom_group_key : 'group1' });
```

### Failed events due to timestamp

If your integration is correct and you are still seeing failed events, review and verify that you are sending all date properties as UTC time format, due to Mixpanel timestamp format requirements.

### Failed events due to messageId

Segment maps the messageId of a Segment event to Mixpanel’s insert\_id value. If you are generating your own messageId, ensure the format complies with Mixpanel’s insert\_id requirements. For more information, see Mixpanel’s [Import Events](#) documentation.

Failing to generate a messageId that complies with Mixpanel’s insert\_id standard might result in a 400 Bad Request error from Mixpanel.

### Why is Boardman, Oregon appearing in my users’ profile location field?

If you are seeing traffic from Boardman or see Segment as the browser, you might be sending server side calls to your Mixpanel (Actions) destination. To correctly populate your users’ profile location field, manually pass the

IP information in the context object from the server.

## Why is the Operating System field empty in Mixpanel?

Mixpanel captures the `Operating System` field from the “OS Name” field in Segment. For Analytics.js sources, ensure that `context.userAgentData.platform` is correctly mapped to the “OS Name” field in your destination mappings. If this mapping is missing or misconfigured, the Operating System field may appear empty in Mixpanel.

This page was last modified: 05 Feb 2025

## 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](#)

