Iterable (Actions) provides the following benefit over Iterable Classic:

- **Transparent data mapping**. The Classic Iterable destination receives data from Segment and converts Segment events to Iterable's format using hard coded mappings that are unable to be adjusted. The Iterable (Actions) destination allows clients to fully define their own mappings of Segment events, ensuring they receive data structured specifically for their needs.

Getting Started

Follow these steps to connect the Iterable (Actions) destination to your Segment sources:

1. Access the Segment web app and click on **Catalog**.
2. In the Catalog, use the search function to find "Iterable". Select the **Iterable (Actions)** destination from the results, and choose which of your sources to connect the destination to.
3. From the Segment web app, navigate to **Connections > Catalog > Destinations**.
4. Click the **Destination Actions** category item in the left navigation.
5. Search for **Iterable (Actions)** and select it.

6. Click **Configure Iterable (Actions)**.
7. Select an existing Source to connect to Iterable (Actions).
8. Complete the Destination Settings as listed below.

Destination Settings

SETTING	DESCRIPTION
API Key	*Required.* To obtain the API Key, go to the Iterable app and naviate to Integrations > API Keys. Create a new API Key with the ‘Server-Side’ type.
Data Center Location	The location where your Iterable data is hosted.

Available Presets

Iterable (Actions) has the following presets:

PRESET NAME	TRIGGER	DEFAULT ACTION
Update Cart Calls	Event type = "track" and event = "Cart Updated"	Cart Updates
Order Completed Calls	Event type = "track" and event = "Order Completed"	Track Purchase
Identify Calls	Event type = "identify"	Upsert User
Track Calls	Event type = "track" and event != "Order Completed" and event != "Cart Updated"	Custom Event

Available Actions

Build your own Mappings. Combine supported triggers with the following Iterable-supported actions:

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

- Custom Event
- Cart Updates
- Track Purchase
- Upsert User

Custom Event

Track a custom event to a user profile

Custom Event is a **Cloud** action. The default Trigger is:
`type = "track" and event != "Order Completed" and event != "Cart Updated"`

Click to show / hide fields

FIELD	DESCRIPTION
Email Address	Type: `STRING` An email address that identifies a user profile in Iterable.

FIELD	DESCRIPTION
User ID	Type: `STRING` A user ID that identifies a user profile in Iterable.
Event name *	Type: `STRING` Name of the event
Event Data Fields	Type: `OBJECT` Additional event properties.
Event ID	Type: `STRING` A unique ID. If an event exists with that id, the event will be updated
Timestamp	Type: `DATETIME` Time the event took place.
Campaign ID	Type: `INTEGER` Iterable campaign the event can be attributed to.
Template ID	Type: `INTEGER` Iterable template the event can be attributed to.
Enable Batching	Type: `BOOLEAN` When enabled, Segment will send data to Iterable in batches of up to 1001

Cart Updates

Updates the shoppingCartItems field on the user profile.

Cart Updates is a **Cloud** action. The default Trigger is: `type = "track" and event == "Cart Updated"`

Click to show / hide fields

FIELD	DESCRIPTION
User Data *	Type: `OBJECT` Updates user data or adds a user if none exists
Cart items *	Type: `OBJECT` Individual items in the cart. Each item must contain `id`, `name`, `price`, and `quantity`. Extra values are added to dataFields.

Track Purchase

Tracks a purchase to Iterable

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

Click to show / hide fields

FIELD	DESCRIPTION
Order ID	Type: `STRING` Similar to `Event ID` in custom events. Iterable recommends mapping `order_id` or `messageId`. If a purchase exists with that id, the purchase will be updated. If none is specified, a new id will automatically be generated by Iterable
User Data *	Type: `OBJECT` Updates user data or adds a user if none exists
Event Data Fields	Type: `OBJECT` Additional event properties.
Cart items *	Type: `OBJECT` Individual items in the cart. Each item must contain `id`, `name`, `price`, and `quantity`. Extra values are added to dataFields.

FIELD	DESCRIPTION
Total *	Type: `NUMBER` Total order amount.
Timestamp	Type: `DATETIME` Time the event took place.
Campaign ID	Type: `INTEGER` Iterable campaign the event can be attributed to.
Template ID	Type: `INTEGER` Iterable template the event can be attributed to.

Upsert User

Creates or updates a user

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

Click to show / hide fields

FIELD	DESCRIPTION
Email Address	Type: `STRING` An email address that identifies a user profile in Iterable.
User ID	Type: `STRING` A user ID that identifies a user profile in Iterable.
User Data Fields	Type: `OBJECT` Data to store on the user profile.
User Phone Number	Type: `STRING` User phone number. Must be a valid phone number including country code. e.g. +14158675309
Merge Nested Objects	Type: `BOOLEAN` If you'd like to merge (rather than overwrite) a user profile's top-level objects with the values provided for them in the request body, set mergeNestedObjects to true.
Enable Batching	Type: `BOOLEAN` When enabled, Segment will send data to Iterable in batches of up to 1001

Important differences from the classic Iterable destination

Since the release of Iterable's Classic Segment destination, Iterable has expanded its support for multiple project types. To determine the appropriate identifier for your project type, please refer to the list of available project types and their respective identifiers found at the following link: Project Types and Unique Identifiers.

Creating or Updating Users

The method by which you identify users depends on the project type you use:

Email-based Projects

In email-based projects, it is necessary to include the email to successfully create a user in Iterable. Once both the email and `userId` have been set in Iterable, the `userId` can be utilized for any future user updates.

UserID-based Projects

For userID-based projects, a unique `userId` is required for creating a user in Iterable. While it is optional to add an email to a userID-based user profile, all subsequent user updates must be performed using the `userId`.

Hybrid Projects**

In hybrid projects, you have the flexibility to choose between using a unique email or a `userId` to create a user in Iterable.

In Iterable's previous classic destination, when making Identify calls, certain context fields were automatically mapped to user profiles. However, this behavior has been changed. Please note that the following context fields are no longer automatically mapped to Iterable user profiles during Identify calls:

- app
- device
- ip
- locale
- page
- timezone

To include these fields in user profiles, pass them as traits with Identify calls. This change offers more control and customization options for managing user data within Iterable.

Additionally, the integration has been updated to support explicit mappings for updating the `phoneNumber` user profile field, as well as support of the `mergeNestedObject` boolean field in user update calls.

Custom Events

In UserID and Hybrid projects, when a passed `userId` doesn't match an existing user, Iterable creates a new user automatically. In email-based projects, tracking a custom event for an unidentified user will not create a user profile.

To ensure proper user profile creation in email-based projects:

- Call the Identify method with both a `userId` and an `email` to create a user profile.
- After you create the user profile, proceed with tracking the custom event for that user.

If you follow this approach, you can guarantee the creation of user profiles and accurately track custom events within Iterable for email-based projects.

Commerce Events

In the classic destination of Iterable, cart updates were associated with Segment's `Product Added` and `Product Removed` events. However, in the Action destination, there have been updates to the default mappings. Now, custom events titled `Cart Updated` are routed to Iterable's Update Cart API.

This page was last modified: 23 Oct 2024

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