



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

Identify, Page and Group calls.

The Segment Tracking Plan feature allows you to validate your expected events against the live events that are delivered to Segment. Violations generate when an event doesn't match the spec'd event in the Tracking Plan.



Segment can infer event data types, but is unable to do so if several data types are sent for a specific category.

Tracking Plans are stored in workspaces and can be connected to one or more Sources.



Segment Consent Preference Updated Event

After setting up a consent category, users of Consent Management see a Segment Consent Preference Updated Event added to all existing Tracking Plans.

Create a Tracking Plan

To create a new Tracking Plan:

1. Contact your Segment account team to enable the Protocols features in your workspace.

2. Once enabled, click **Protocols** in the left bar navigation.

3. Click **New Tracking Plan**.

4. Add events, properties, traits and filters in the Tracking Plan editor.

You'll see an option to import events and traits to your Tracking Plan that your source received in the last 24 hours, 7 days or 30 days. This option is great if you want to get started with your current events.



Consent Management users see the Segment Consent Preference Updated event on new Tracking Plans

If you are a Consent Management user and have created at least one consent category, Segment automatically adds the [Segment Consent Preference Updated event](#) to all new Tracking Plans.

Copy a Tracking Plan

To create a copy of an existing Tracking Plan:

1. Click **Protocols** in the left navigation bar.

2. On the row of the Tracking Plan you want to copy, open the contextual menu(...), and select **Duplicate Tracking Plan**.

3. Enter a name for the new Tracking Plan instance, and click **Duplicate**.

Download a Tracking Plan

To download a Tracking Plan:

1. Click **Protocols** in the left navigation bar.

2. On the row of the Tracking Plan you want to download, open the contextual menu(...), and select **Download Tracking Plan**.

3. A toast pops up on the top of the page, with the message *"Your file is processing. When your file is ready it will be available to download from the Download History page."*

4. Open the Download History page by clicking the link in the toast or clicking the **Download History** tab in the top navigation bar.

5. Once the file status column indicates that the download was successful, click the link in the File column to download your CSV to your computer. If the file status column shows the download has failed, return to the Tracking Plan Overview page or the Tracking Plan page and try the download again.

The Tracking Plan CSV name has the following format:

`workspaceSlug-trackingPlanName--yyyy-mm-dd--hh-mm-utc`

The columns in the Tracking Plan CSV file corresponds to the Tracking Plan UI options. For example:

Allowed Property Values: In the Tracking Plan UI, when the property type is 'String', you have the option to add a list of permitted values.

Enum Values: When the property type is 'Enum', you have the option to add a list of permitted values.

Once you've downloaded a Tracking Plan, you can [upload it](#) as a template for a new Tracking Plan or use it to make changes to an existing Tracking Plan.

Upload a Tracking Plan

You can create a Tracking Plan or make changes to an existing Tracking Plan by uploading a CSV that contains the rules and events you'd like to track. Segment provides a Tracking Plan template file that you can download

during the import process, or you can [download an existing Tracking Plan](#) to use as your template.



Tracking Plan CSV requirements

Tracking Plan CSV files uploaded to Segment must be smaller than 15 mb and contain one header row and one or more rows of data. Tracking Plans CSVs must also have fewer than 100,000 rows and 2,000 rules.

Create a new Tracking Plan

To create a new Tracking Plan by uploading a CSV file:

1. Click **Protocols** in the left navigation bar.

2. Click **New Tracking Plan**.

3. Click the **Import...** button and select **From CSV**.

4. Download the Tracking Plan template CSV and fill in the template file with your new Tracking Plan rules, or [download an existing Tracking Plan](#).

5. Once you've filled in the provided template or made changes to your downloaded Tracking Plan, add your CSV file to the file uploader and click **Upload**.

After uploading your CSV file, you are redirected to the Upload & Download History page while the upload is in progress. If the CSV upload fails, you'll be able to either view the error directly in the Reports column on the Upload & Download History page or download the `error_report.csv` file that corresponds to the Tracking Plan you uploaded.



Tracking Plans created by an uploaded file are reflected in the [Audit Trail](#) and [Tracking Plan changelog](#). If you are a Consent Management user and have created at least one consent category, Segment automatically adds the [Segment Consent Preference Updated event](#) to all new Tracking Plans.

Update an existing Tracking Plan



Tracking Plans with imported libraries cannot be changed using the Upload a Tracking Plan method

If you have a Tracking Plan with imported libraries, you must make changes to your Tracking Plan in the Segment app.

To update a Tracking Plan by uploading a CSV file:

1. Click **Protocols** in the left navigation bar.

2. On the row of the Tracking Plan you want to edit, open the contextual menu(...) and select **View Tracking Plan**.

3. Select **Edit Tracking Plan**.

4. Click the **Import...** button and select **From CSV**.

5. Download the Tracking Plan template CSV and fill in the template file with your new Tracking Plan rules, or [download an existing Tracking Plan](#).

6. Once you've filled in the provided template or made changes to your downloaded Tracking Plan, add your CSV file to the file uploader and click **Upload**.

After uploading your CSV file, you are redirected to the Upload & Download History page while the upload is in progress. If the CSV upload fails, you'll be able to either view the error directly in the Reports column on the Upload & Download History page or download the `error_report.csv` file that corresponds to the Tracking Plan you

uploaded.



Any changes made to a Tracking Plan using an uploaded file are reflected in the [Audit Trail](#) and [Tracking Plan changelog](#).

Delete a Tracking Plan



Deleting a Tracking Plan requires Workspace Owner or Tracking Plan Admin permissions

You must have Workspace Owner or Tracking Plan Admin roles to delete a Tracking Plan. For more information about roles in Segment, see the [Roles documentation](#).

To delete a Tracking Plan:

1. Open the Segment app and click **Protocols**.
2. On the row of the Tracking Plan you want to delete, open the contextual menu(...) and select **Delete Tracking Plan...**
3. On the "Delete Tracking Plan" popup, click **Delete**.

Edit a Tracking Plan

The Tracking Plan editor is organized as a spreadsheet to help you add new events and properties, and edit the relevant fields for each. Like a spreadsheet, you can navigate across cells in a single event with your arrow keys and press enter to edit a cell.

COLUMN NAME	DETAILS
Name	Specify the name of your event or property.
Description	Enter a description for your event or property. These descriptions are helpful for both engineers instrumenting Segment and consumers of the data.
Status	Specify whether a property is required or optional. You can't require a Track call because Segment is unable to verify when a Track call should be fired.
Data Type	Specify the data type of the property. Data type options include any, array, object, boolean, integer, number, string, null, Date time . Note: Date time is required to be in ISO-8601 format
Permitted Values	Enter simple regular expressions to validate property values. This works when a property data type is set to string . For example, you can add pipe delimited strings to the regex column to generate violations when a property value does not match fall, winter or spring.



The Status, Data Type, and Permitted Values columns appear as you add a Track call property.

Add a new Track call

To add a new Track call:

1. Click **Add Event** to add a new row.
2. Click into the row to add an event name and description. The event name strictly validates the name passed in your Track calls. Casing, spacing and spelling matter.

Add a Track call property

To add a Track call property:

- 1 Click on the **(+)** next to the event name to add a new row below the event name.
- 2 Click into the row to add the property name and also specify the description, status, data type and permitted values when applicable.
- 3 You can use your keyboard arrow and enter keys to navigate across the cells, or use your mouse.

Add a Track call object or array property

Segment supports object and array data types in the Tracking Plan editor. These complex data structures have limited use cases and should be used sparingly as some destinations aren't able to ingest the data structures. To add an object or array:

- 1 Create a new property row and set the Data Type to **Object** or **Array**.
- 2 Click the **(+)** next to the property name to add key value pairs in the object, or objects to an array of objects.



When creating array properties in your Tracking Plan, add the **items** nested property, denoted by the name of the array property with a **.\$** suffix, to ensure that the nested property is marked as planned in the Source Schema.

Add Identify or Group traits

You can define which traits you expect to see passed in Identify or Group calls like how you would add Track calls to the Tracking Plan. Navigate to the **Identify** or **Group** tab in your Tracking Plan and click the **(+)** button to add a new trait.

It's best to keep traits optional because Identify and Group are often called and pass only *new or changed* traits, because Segment's client-side libraries (analytics.js, Swift, Kotlin) cache traits in local storage. See the [Identify Best Practices](#) to learn more.

Remove a source from your Tracking Plan



Removing a source from a Tracking Plan requires Workspace Owner or Tracking Plan Admin permissions

You must have Workspace Owner or Tracking Plan Admin roles to remove a source from a Tracking Plan. For more information about roles in Segment, see the [Roles documentation](#).

To remove a source from your Tracking Plan:

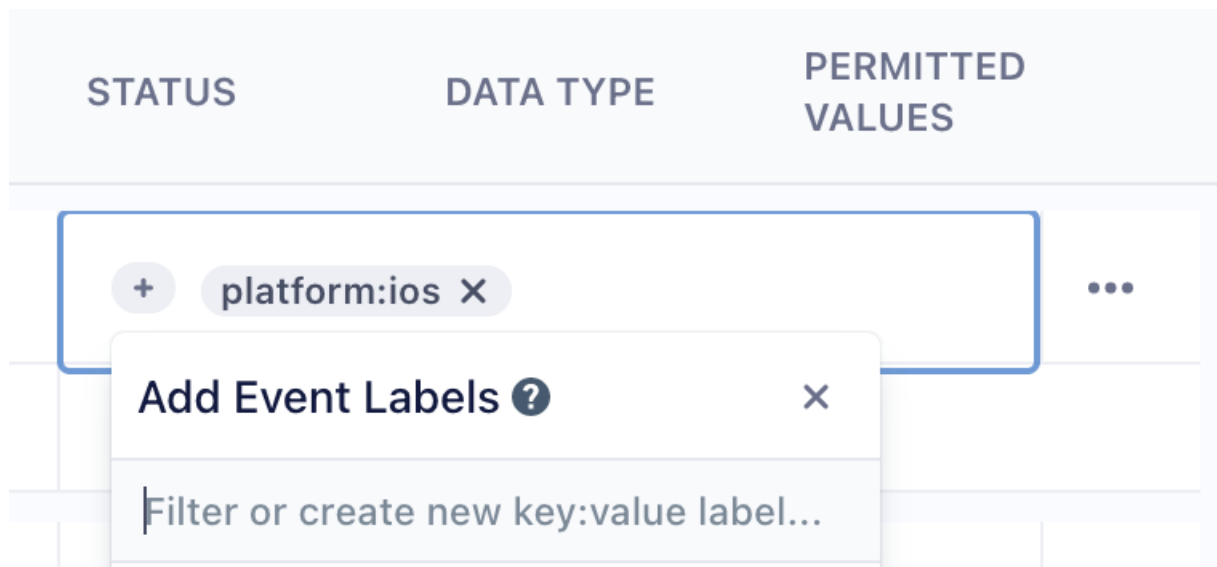
- 1 Click **Protocols** in the left navigation bar.
- 2 Find the Tracking Plan you'd like to remove a source from, then select the source icon in the Connected Sources column.
- 3 On the list of sources connected to your Tracking Plan, find the source you'd like to remove from your Tracking Plan and click **Disconnect**.
- 4 Review the "Disconnect source from Tracking Plan" popup and click **Disconnect**.

Add a label

You can apply **key:value** labels to each event to help organize your Tracking Plan. These labels are helpful when multiple teams are managing a single Tracking Plan, or if you want to specify a priority, platform, product, or similar meta-data for each event. You can filter by label from the Tracking Plan, Schema, Data Validation and Violations Summary views.

For consistency purposes, it's best that you create a standard way of labeling events and share it with all parts

of your organization that will use Segment.



Note: Tracking Plan Labels are only available for Track and Page events.

Filter events in the Tracking Plan

You can filter the Tracking Plan events by keyword or by label. The applied filter generates a permanent link so you can share specific events with teammates. Label filters also persist after you leave the Tracking Plan.

Edit underlying JSON Schema

Protocols Tracking Plans use [JSON Schemas](#) to validate Segment event payloads. To support a broader range of validation use-cases, Segment lets you to edit your underlying JSON schema.



Editing a JSON schema requires technical expertise. The [JSON schema documentation](#) and [JSON schema validator](#) are helpful resources you can use.

You can edit the JSON schema for each Track event listed in the Tracking Plan, and a common JSON schema definition that applies across all events.

Track event JSON schema

Each Track event in the Tracking Plan has a separate JSON schema definition to validate the properties in that event. To edit, click on the overflow menu next to each event row in the Tracking Plan.



Advanced edits to the JSON schema are not visible in the Tracking Plan and make it harder for other users to understand the validation logic. Be sure to communicate to any other Protocols users that you are making changes in the validation logic.

Common JSON schema

The Tracking Plan also uses a common JSON schema definition that applies to the entire payload of every event sent from sources connected to the Tracking Plan.

The common JSON schema definition is unique for each Tracking Plan. An example use of this feature is to validate that all Track, Identify and Page events sent to Segment include a `context.device.advertisingId` property. This validation ensures that every Segment call has a `userId`, `anonymousId`, and `context` object with a nested

"device": { "advertisingId": "e23sfsdf"} object.

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "type": "object",
  "properties": {
    "context": {
      "type": "object",
      "properties": {
        "device": {
          "type": "object",
          "properties": {
            "advertisingId": { "type": "string" }
          },
          "required": ["advertisingId"]
        }
      },
      "required": ["device"]
    },
    "anonymousId": {
      "type": "string"
    },
    "userId": {
      "type": "string"
    },
    "properties": {
      "type": "object",
      "properties": {
        "primary_business_unit": {
          "type": "string"
        }
      },
      "required": ["primary_business_unit"]
    },
    "traits": {
      "type": "object",
      "properties": {
        "first_name": {
          "type": "string"
        },
        "last_name": {
          "type": "string"
        },
        "email": {
          "type": "string"
        }
      },
      "required": ["email"]
    }
  },
  "required": ["context", "anonymousId", "userId", "traits"]
}
```

To edit the common JSON schema:

- 1 Click the (...) at the top of the Tracking Plan editor and select **Edit Common JSON Schema**.
- 2 Enter your new JSON schema and click **Update JSON**.
- 3 Once you've saved and merged your JSON schema changes, go to the **Settings** tab for your source.
- 4 Click **Schema Configuration** in the navigation and go to the **Advanced Blocking Controls** section to define specific blocking behavior for common JSON schema violations.

To edit the common JSON schema using the Public API, you'll need to add your new JSON schema under the "global": object.



Negative lookahead regexes (?!) aren't supported. This means you can't use regex to prevent matches with a specific following character or expression. But, you can use **not** in the regex of your JSON schema to generate violations when a property key or value doesn't match the provided regex pattern.



Specifying data type

Property or trait data type should adhere to the [data types defined by JSON schema](#). Data type names must be lower-cased as specified in JSON schema. Date/time properties should be represented as a **string** type with **format** **keyword** (for example: "format": "date-time").



Blocking data

JSON schema violation event blocking is **only** supported in cloud-mode Destinations. See [the Customize your schema controls docs](#) for more information on blocking data.

Extend the Tracking Plan

Some customers prefer to manage the Tracking Plan with outside tools and resources. See the [APIs and extensions](#) section to learn more.

Tracking Plan Event Versioning

Segment offers Tracking Plan Event Versioning if you use Protocols to manage mobile sources, or to help you centrally manage a Tracking Plan for multiple teams. With Event Versioning, you can create multiple versions of an event definition, and validate events using a version key included in the Track event payload.

This can be helpful for mobile developers who might have several released versions of their app sending data at the same time. For example, a new mobile app release might add a new required property to an event like `Order Completed`. In this scenario, if you updated the Tracking Plan, all `Order Completed` events from your old mobile app versions would be invalid, because some customers won't have updated to the latest version yet. Instead, with event versioning, you can allow validation of both the old and new versions of an event at the same time. When you're ready to deprecate those old event versions, you can delete the version in your Tracking Plan.

For example, say you want to add `subtotal` as a required property to your `Order Completed` event. You would start by adding the required property to the event in the Tracking Plan as shown in the example below.

▼	<input type="checkbox"/>	Order Completed	⊕	When user completes an order	+	
	<input type="checkbox"/>	products	⊕	Array of product objects	required ▼	array ▼
	<input type="checkbox"/>	items	⊕	Description...	optional	object ▼
	<input type="checkbox"/>	order_id		Backend order id	required ▼	string ▼
	<input type="checkbox"/>	revenue		Inclusive of shipping and tax	required ▼	number ▼

▼	<input type="checkbox"/>	Order Completed	⊕	When user completes an order	+	
	<input type="checkbox"/>	subtotal		Exclusive of shipping + tax	required ▼	number ▼
	<input type="checkbox"/>	products	⊕	Array of product objects	required ▼	array ▼
	<input type="checkbox"/>	items	⊕	Description...	optional	object ▼
	<input type="checkbox"/>	order_id		Backend order id	required ▼	string ▼
	<input type="checkbox"/>	revenue		Inclusive of shipping and tax	required ▼	number ▼

Create a new event version

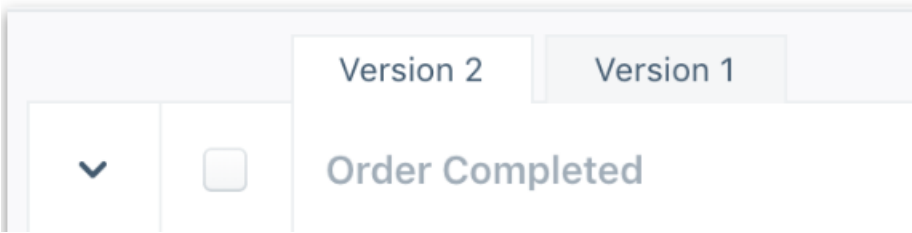
With event versioning, you can now create multiple versions of the event definition as shown in the example

below. To create a new event version, click into the overflow menu for an event and select **Add Event Version**.

		Version 2	Version 1				
▼	<input type="checkbox"/>	Order Completed	⊕	When user completes an order	+		
	<input type="checkbox"/>	subtotal		Exclusive of shipping + tax	required ▼	number	<div>View Code Snippet... Edit JSON Schema... Duplicate Event Add Event Version</div>
	<input type="checkbox"/>	products	⊕	Array of products including product...	required ▼	array	
	<input type="checkbox"/>	items	⊕	Description...	optional	object	
	<input type="checkbox"/>	product_id		Product id	optional ▼	string	
	<input type="checkbox"/>	product_name		Product name	optional ▼	string ▼	Enter regular expression...
	<input type="checkbox"/>	order_id		Backend order id	required ▼	string ▼	Enter regular expression...
	<input type="checkbox"/>	revenue		Inclusive of shipping + tax	required ▼	number ▼	

Dynamically validate Track events against an event version

To ensure the Track events you send to a Segment source validate against the correct event version, you need to instrument your events to include a `context.protocols.event_version` key and version value. The version value must pass as an integer, and should match the number shown in the Tracking Plan version tab. In the example below, the version number would be **2**.



Next, add the event version number to the context object. For [analytics.js](#) Track calls, you would instrument the event as in the example below. Note how the JSON objects for `context`, `protocols`, and `event_version` are nested.

```
analytics.track('Order Completed', {
  subtotal: 23,
  products: [{
    product_name: 'Air Balloon',
    product_id: '32rd9jfs'
  }],
  order_id: '2df90eiwc9wjec',
  revenue: 33
}, {
  context: {
    protocols: {
      event_version: 2
    }
  }
});
```

Note: Protocols validate events against the oldest event version in the Tracking Plan for event payloads that are 1) missing the `context.protocols.event_version` key, or 2) contain an invalid/undefined event version (ex: `event_version:3.2`).

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

