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

Device-mode	Cloud-mode
✓ Web	✓ Web
⊘ Mobile	✓ Mobile
⊘ Server	✓ Server

Eloqua is Oracle’s B2B Cross-Channel Marketing solution, and it operates by tracking the identities and event actions of users that interact with your properties.

With Segment, all the customer data you’re collecting from your websites, mobile apps, or servers sends directly into Eloqua, using your existing Segment implementation. This means you don’t need to write any custom code or create duplicate event tracking, but you can start using Eloqua with a simplified setup effort.

Getting Started

1. From the Segment web app, navigate to **Connections > Catalog** and select the **Destinations** tab within the

catalog.

2. Search for **Eloqua** and select the destination.
3. Click **Configure Eloqua**.
4. Select the web source that will send data to Eloqua and follow the steps to name your destination.
5. On the **Settings** tab, input your **Company Name**, **Password**, **Site ID**, **Username**, and configure **Other Settings**.
6. Enable the destination.

When you enable Eloqua in your Segment integrations page, Segment automatically loads Eloqua's JavaScript libraries onto your web pages. For mobile and server-side events, Segment routes your event data directly through Eloqua's servers, eliminating the need for any Eloqua code to be loaded in your applications.

Page

Client-side page-view tracking is achieved using an integration with the Eloqua Asynchronous Visitor Tracking Script.

Page tracking with Eloqua is, by default, achieved with a third party cookie. This cookie is generated upon successful completion of an Eloqua form. Once a user submits one of these forms, Segment `page` events will start populating as Website Visits in their Eloqua Contact profile.

Identify

Segment integrates with Eloqua's REST API to support creating or updating Contacts in Eloqua following events that do not use an Eloqua form.

Upon invocation of a server-side `identify`, Segment will by default map the Segment `identify` trait on the left to the Eloqua field on the right:

SEGMENT TRAIT	ELOQUA FIELD
`email`	Email
`firstName`	First Name
`lastName`	Last Name
`street` *or* `address.street`	Address
`city` *or* `address.city`	City
`country` *or* `address.country`	Country
`title`	Title

Follow the Segment spec to ensure proper mapping of these fields from Segment `identify` traits: /docs/connections/spec/identify/#traits.

In addition, Segment supports mapping custom `identify` traits to Eloqua custom object fields. To do so, you can set up mappings in the settings for your Eloqua destination in the Segment UI.

Group

Segment integrates with Eloqua's REST API to support creating or updating Accounts in Eloqua. By default, Segment does not send `group` events downstream to Eloqua, so to get started, enable "Create or Update Account on Group" in your destination settings.

Upon invocation of a server-side `group` event, Segment will by default map the Segment `group` trait on the left to the Eloqua field on the right:

SEGMENT TRAIT	ELOQUA FIELD
`name` + `groupId`	Company Name
`street` *or* `address.street`	Address 1
`city` *or* `address.city`	City
`country` *or* `address.country`	Country
`city` *or* `address.city`	City
`country` *or* `address.country`	Country
`phone`	Business Phone

Segment concatenates `traits.name` and `groupId` to Company Name to ensure uniqueness. Therefore, in order to successfully create or update an Account, the Segment `group` event needs to include the Account's name and groupId. If you set up a custom Account field called Company in Eloqua, Segment will automatically map the name of the Account to that field.

Follow Segment's group spec to ensure proper mapping of these fields from Segment `group` traits.

In addition, Segment supports mapping custom `group` traits to Eloqua custom object fields. To do so, you can set up mappings in the settings for your Eloqua destination in the Segment UI.

Mapping custom traits to Eloqua Accounts and Contacts

First, configure the custom Account fields or custom Contact fields in your Eloqua dashboard. Read how to set those up for Contacts here and for Accounts here.

Once you are set up in Eloqua, you are ready to map custom traits to your Contacts and Accounts. Next, provide a mapping in your destination settings that specifies the Segment trait names and the corresponding custom Account or Contact fields to associate them with in Eloqua.

Segment's custom trait mapping will match exactly what you input into your Segment settings. For example, if you have a field with a display name called `Account Rating` in Eloqua, and a property called `AccountRating` in your Segment events, you should enter `AccountRating: Account Rating` as the mapping in the "Map Custom Traits to Accounts" setting in the Segment UI.

If a match between a custom Account or Contact field has incompatible data types, Eloqua will return an error for the entire request.

Track

Segment `track` events trigger the creation of Eloqua Custom Object records and associate them with a specific Contact.

To get started, provide a mapping in your destination settings specifying the Segment event names with the corresponding Custom Object you would like to associate it with in Eloqua.

Segment will also map the properties of `track` events with Custom Object fields of the same name. The integration does an automatic case- and formatting-insensitive match so that if you have a field called `Account Type` in Eloqua and a property called `AccountType` in your Segment event, the mapping will get handled.

ⓘ
Track event mapping limitations

Each `track` event can only be mapped to a single custom object. Segment doesn't support mapping a single `track` event to multiple custom objects.

For `track` event properties you intend to send to Eloqua as Custom Object fields, make sure the value of the data type sent to Segment matches the data type specified in your Eloqua dashboard. If a Custom Object field data type does not match the data type specified in Eloqua, Eloqua will not create a Custom Object, and will return an error for the request.

Segment identifies Eloqua Custom Object Fields with `date` data types, and converts date objects to Unix date strings in seconds, before sending the payload to Eloqua.

ELOQUA DATA TYPE	COMMON DATA TYPE	EXAMPLE VALUE
Number	Integer	31415
Text	String (max 250 characters)	'31415'
Large Text	String (max 32000 characters)	'31415'
Date/Time	String (Unix date string in seconds)	'1543861960'
Numeric	Float (up to 4 decimal places)	3.1415

User Email

By default, Eloqua uses email address as a Contact ID. Therefore, in order to successfully associate a Contact with a Custom Object, the Segment `track` event needs to include the user's email as a trait or integration-specific property:

```
analytics.track({
  event: 'Account Created',
  context: {
    traits: {
      email: 'billbrasky@segment.com'
    }
  }
})
```

```
analytics.track({
  event: 'Account Created',
  integrations: {
    Eloqua: {
      email: 'billbrasky@segment.com'
    }
  }
})
```

Creating an Eloqua Contact on `track`

By default, Segment creates an Eloqua Contact on `identify` and only associates Contacts with Custom Objects on Segment `track`.

Enable the `Create Contact on Track` Segment setting to both create Eloqua Contacts and associate the Contact with an Eloqua Custom Object when you invoke a Segment `track` event.

Note that, to help reduce the number of API calls Segment sends to Eloqua, enabling this setting will prevent `identify` events from flowing to Eloqua, so user traits should be included as `context.traits` in `track` events.

As an example, the following event would create or update Contact `billbrasky@segment.com` with the values passed in `context.traits`, then associate the Bill with Eloqua Custom Object `Account Created`.

```
analytics.track({
  event: 'Account Created',
  properties: {
    referrer: 'Google Ad'
  },
  integrations: {
    Eloqua: {
      email: 'billbrasky@segment.com'
    }
  },
  context: {
    traits: {
      address: '123 Market Street',
      city: 'San Francisco'
    }
  }
})
```

Creating an Eloqua Account on `track`

If you've enabled the "Create or Update Account on Group" setting, Segment `track` events will by default create an Eloqua Account if you provide a `name` and `groupId` in the `context.traits` field. Both fields must be present to create or update an Account downstream in Eloqua.

```
analytics.track({
  event: 'Account Created',
  integrations: {
    Eloqua: {
      email: 'billbrasky@segment.com'
    }
  },
  context: {
    traits: {
      name: "Segment.com",
      groupId: "1234567"
  }
})
```

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 Eloqua 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

SETTING	DESCRIPTION
Company Name *(required)*	`string`. Enter your login company name
Create or Update Contact on Track	`boolean`, defaults to `FALSE` . *Server Side Only*: Enable this setting to create or update an Eloqua `contact` with traits from your `track` events' `context.traits` object. *Note that when this setting is enabled, the destination will only process `track` events and will reject `identify` events.* This is to reduce the number of potentially repetitive API calls sent downstream to Eloqua.
Map Track Events to Custom Objects	`text-map`, defaults to {}. Please input the Segment event names on the left and their corresponding Eloqua Custom Object names on the right. This mapping is required for all events you would like in Eloqua as Custom Objects. **Note:** If you have set up Custom Object Fields in Eloqua, please ensure the corresponding Segment payload property values are of the same data type specified in Eloqua's dashboard. Eloqua will reject any event containing a Custom Object Field with an incorrect data type. Segment automatically attempts to match property names to Custom Object Field names in Eloqua.
Map Custom Traits to Accounts	`text-map`, defaults to {}. Please input the Segment trait names on the left and their corresponding Eloqua Custom Account Field names on the right. The traits must be set up in the Eloqua dashboard prior to instantiating this mapping. **Note:** If you have set up Custom Account Fields in Eloqua, please ensure the corresponding Segment payload property values are of the same data type specified in Eloqua's dashboard. Eloqua will reject any event containing a Custom Account Field with an incorrect data type.
Map Custom Traits to Contacts	`text-map`, defaults to {}. Please input the Segment trait names on the left and their corresponding Eloqua Custom Contact Field names on the right. The traits must be set up in the Eloqua dashboard prior to instantiating this mapping. **Note:** If you have set up Custom Contact Fields in Eloqua, please ensure the corresponding Segment payload property values are of the same data type specified in Eloqua's dashboard. Eloqua will reject any event containing a Custom Contact Field with an incorrect data type.
Password *(required)*	`password`. Enter your login passord
Create or Update Account on Group	`boolean`, defaults to `FALSE` . *Server Side Only*: Enable this setting to create or update an Eloqua account on a `group` event.
Site ID *(required)*	`string`. Enter your Site ID
Username *(required)*	`string`. Enter your login username

This page was last modified: 27 Oct 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.