



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

Segment creates and merges user profiles based on a space's Identity Resolution configuration. Segment searches for identifiers such as `userId`, `anonymousId`, and `email` on incoming events and matches them to existing profiles or creates new profiles. These identifiers display in the Identities tab of a User Profile in the Profile explorer.





Navigate to **Unify > Profile explorer** to view identities attached to a profile, along with custom traits, event history, and more.

JD

Jane Doe

jane.doe@gmail.com

TraitsAudiencesEventsIdentitiesAssociated Accounts

	email From Mobile (segkicks) at Nov 15, 2019	jane.doe@gmail.com
	user_id From Mobile (segkicks) at Nov 15, 2019	abc123def456
	ios.id From Mobile (segkicks) at Nov 15, 2019	ios_abc123
	anonymous_id From Mobile (segkicks) at Nov 15, 2019	anon_123

Flat matching logic

After receiving a new event, Segment looks for profiles that match any of the identifiers on the event.

Based on the existence of a match, one of three actions can occur:

- 1: Create a new profile** When there are no pre-existing profiles that have matching identifiers to the event, Segment creates a new user profile.
- 2: Add to existing profile** When there is one profile that matches all identifiers in an event, Segment attempts to map the traits, identifiers, and events on the call to that existing profile. If there is an excess of any identifier on the final profile, Segment defers to the Identity Resolution rules outlined below.
- 3: Merge existing profiles** When there are multiple profiles that match the identifiers in an event, Segment checks the Identity Resolution rules outlined below, and attempts to merge profiles.

Identity Resolution settings

Identity Admins should first configure Identity Resolution Settings to protect the identity graph from inaccurate merges and user profiles.

During the space creation process, the first step is to choose an Identity Resolution configuration. If this is your first space, you have the option to choose a Segment-suggested Out-of-the-Box configuration or a custom Identity Resolution setup. All other spaces have a third option of importing settings from a different space.

Identity rules determine which identifiers are collected and how they are prioritized to build out user profiles surfaced in audiences and computed traits. Please select your preferred set up method below:

Import from an Existing Space *(Recommended)*



Apply the same identity rules that you have already set and validated in an existing space.

Out-of-the-Box

We will help you find a "best fit" set of identity rules based on your use case and input.

Custom Rules

If you are familiar with using identity configuration rules you may create a custom set within Identity settings.

Continue

Out-of-the-box

For most first-time users, Segment recommends that you use the out-of-the-box configuration and answer a short series of questions for a best-fit setup for your use-case.

If you have custom unique identifiers or don't have a canonical `user_id`, you're automatically redirected to the Identity Resolution Settings page to complete your setup.

Custom rules

If you're familiar with identity or have custom identifiers, Segment recommends that you select Custom Rules.

Segment redirects you to the Identity Resolution Settings page where you can add Default Identifiers or Custom Identifiers.

Segment's 11 default are:

EXTERNAL ID TYPE

MESSAGE LOCATION IN TRACK OR IDENTIFY CALL

EXTERNAL ID TYPE	MESSAGE LOCATION IN TRACK OR IDENTIFY CALL
user_id	userId
email	traits.email or context.traits.email
android.id	context.device.id when context.device.type = 'android'
android.idfa	context.device.advertisingId when context.device.type = 'android' AND context.device.adTrackingEnabled = true
android.push_token	context.device.token when context.device.type = 'android'
anonymous_id	anonymousId
ga_client_id	context.integrations['Google Analytics'].clientId when explicitly captured by users
group_id	groupId
ios.id	context.device.id when context.device.type = 'ios'
ios.idfa	context.device.advertisingId when context.device.type = 'ios' AND context.device.adTrackingEnabled = true
ios.push_token	context.device.token when context.device.type = 'ios'

You can also provide a trait or property key to match on to add custom identifiers. You can preview the locations where Segment looks for the identifier. Segment accepts both camelCase and snake_case for context.traits, traits, and properties, but accepts lowercase types for identifiers only in the context.externalIds object.

Custom Identifier



Trait / Property key to match on

app_id

Preview message locations

```
{
  "type": "identify"/"group",
  "traits": {
    "appId":
    "app_id":
  },
  "context": {
    "externalIds": [
      { "type": "app_id", "Id" }
    ],
    "traits": {
      "appId":
      "app_id":
    }
  }
}

{
  "type": "track"/"page"/"s
  "properties": {
    "appId":
    "app_id":
  },
  "context": {
    "externalIds":[
      { "type": "app_id",
    ],
    "traits": {
      "appId":
      "app_id":
    }
  }
}
```

Value limit

5

Select frequency

Ever

The maximum number of unique values for app_id that a profile can accept.

Cancel

Add new identifier

Blocked values

Segment recommends that you proactively prevent using certain values as identifiers. While these values remain in the payload on the event itself, it is not promoted to an identifier Segment uses to determine user profiles.

This is important when developers have a hard-coded value for fields like user_id during QA or development that then erroneously make it to production. This may cause hundreds of profiles to merge incorrectly and can have costly consequences if these spaces already feed data into a production email marketing tool or push notification tool downstream.

In the past, Segment has seen certain default values that cause large amounts of profiles to merge incorrectly. Segment suggests that for every identifier, customers opt into automatically blocking the following suggested values:

VALUE	TYPE
Zeroes and Dashes (^[0-]*\$)	Pattern (REGEX)
-1	Exact Match

VALUE	TYPE
null	Exact Match
anonymous	Exact Match

Blocked Values

Specify values for user_id that should be ignored by the identity algorithm.

Blocking values here will not drop entire events, only specific identifiers. You should use this feature if you have certain known, bad identifier values in your instrumentation that you'd like to prevent from appearing on profiles.

Add Custom Value

Add Recommended Value

Blocked Value	RECOMMENDED	Type	
-1	<input checked="" type="checkbox"/> Zeroes and dashes	Pattern	atch ×
anonymous	<input checked="" type="checkbox"/> -1	Match	atch ×
	<input checked="" type="checkbox"/> anonymous	Match	atch ×
null	<input checked="" type="checkbox"/> null	Match	atch ×
Zeroes and dashes		Pattern	atch ×

Before sending data through, Segment also recommends that you add any default hard-coded values that your team uses during the development process, such as `void` or `abc123`.

Limit

Identity Admins can specify the total number of values allowed per identifier type on a profile during a certain period. For example, in the image below, the `anonymous_id` field has a limit of **5 Weekly**.

Value Limit

Weekly

The maximum number of unique values for anonymous_id per user profile can accept.

Blocked Values

Specify values for anonymous_id that should be blocked by the deduplication algorithm.

Blocking values here will not drop entire events, but will drop the identifiers. You should use this feature if you have certain known, bad identifiers that you'd like to prevent from appearing on profiles.

Daily

✓ Weekly

Monthly

Annually

Ever

This will vary depending on how companies define a user today. In most cases, companies rely on user_id to distinguish user profiles and Segment defaults to the following configurations:

IDENTIFIER	LIMIT
user_id	1
all other identifiers	5

Specific cases may deviate from this default. For example, a case where a user can have more than one user_id but one email, like when shopify_id and an internal UUID define a user. In this case, an example configuration may be:

IDENTIFIER	LIMIT
email	1
user_id	2
all other identifiers	5

When you choose the limit on an identifier, ask the following questions about each of the identifiers you send to Segment:

1. Is it an immutable ID? An immutable ID, such as user_id, should have 1 ever per user profile.
2. Is it a constantly changing ID? A constantly changing ID, such as anonymous_id or ga_client_id, should have a short sliding window, such as **5 weekly** or **5 monthly**, depending on how often your application automatically logs out the user.
3. Is it an ID that updates on a yearly basis? Most customers will have around five emails or devices at any one time, but can update these over time. For identifiers like email, android.id, or ios.id, Segment recommends a longer limit like **5 annually**.

Priority

Segment considers the priority of an identifier once that identifier exceeds the limit on the final profile.

For example, consider a Segment space with the following Identity Resolution configurations:

IDENTIFIER	LIMIT	PRIORITY
user_id	1	1
email	5	2
anonymous_id	5	3

A profile already exists with `user_id` **abc123** and `email` **jane@example1.com**. A new event comes in with new `user_id` **abc456** but the same `email` **jane@example1.com**.

If this event maps to this profile, the resulting profile would then contain two `user_id` values and one `email`. Given that `user_id` has a limit of 1, this exceeds the limit of that identifier. As a result, Segment checks the priority of the `user_id` identifier. Because `email` and `user_id` are the two identifiers on the event and `email` ranks lower than `user_id`, Segment demotes `email` as an identifier on the incoming event and tries again.

At this point, the event searches for any profiles that match just the identifier `user_id` **abc456**. Now there are no existing profiles with this identifier, so Segment creates a new profile with `user_id` **abc456**.

By default, Segment explicitly orders `user_id` and `email` as rank 1 and 2, respectively. All other identifiers are in alphabetical order beginning from rank 3. This means that if the identifiers sent with events flowing into Segment are `user_id`, `email`, `anonymous_id`, and `ga_client_id`, the rank would be as follows:

IDENTIFIER	PRIORITY
user_id	1
email	2
anonymous_id	3
ga_client_id	4

If a new `android.id` identifier appeared without first giving it explicit order, the order would automatically reshuffle to:

IDENTIFIER	PRIORITY
user_id	1
email	2
android.id	3
anonymous_id	4
ga_client_id	5

If you require an explicit order for all identifiers, configure this in the Identity Resolution Settings page before sending in events.

Identity Resolution Configuration

 Editing

 Edit Priority

 Add Identifier

Priority	Identifier	Seen	Value Limit	Blocked Values	Identifier Type
1	user_id	•	1	—	Default
2	email	•	5	—	Default
 2	android.id	•	5	—	Default
 2	android.idfa	•	5	—	Default
 2	android.push_token	•	5	—	Default
 2	anonymous_id	•	5	—	Default
 2	braze_id	•	5	—	Default
 2	cross_domain_id	•	5	—	Default

When choosing the priority of your identifier, ask the following questions about each of the identifiers you send to Segment:

1 Is it an immutable ID? Give immutable IDs, such as user_id, highest priority.

2 Are they unique IDs? Give Unique IDs such as email higher priority than possibly shared identifiers like android.id or ios.id.

3 Does it temporarily identify a user? Identifiers such as anonymous_id, ios.idfa, and ga_client_id are constantly updated or expired for a user. Generally speaking, rank these lower than identifiers that permanently identify a user.

Importing from an existing space

This option is available to new spaces after you create an initial Dev space. Segment recommends this option when identity settings are validated as correct in the initial Dev space and should be copied into the Production space.

You can review the identifiers, priorities, limits, and blocked values before you complete the import.

Identity Rules Set Up



For your Prod space, you may import identity rules from an existing space. Please ensure that these rules have been tested and validated in a space prior to import.

Import From

Peter Test Space



Priority	Identifier	Value Limit	Blocked Values
1	user_id	1	<i>Zeroes and dashes, anonymous</i>
2	email	5/Annually	<i>Zeroes and dashes, anonymous</i>
3	group_id	5/Annually	<i>Zeroes and dashes, anonymous</i>
4	ios.idfa	2/Weekly	<i>Zeroes and dashes, anonymous</i>
5	android.idfa	2/Annually	<i>Zeroes and dashes, anonymous</i>
6	android.id	2/Annually	<i>Zeroes and dashes, anonymous</i>
7	ios.id	2/Annually	<i>Zeroes and dashes, anonymous</i>
8	anonymous_id	5/Weekly	<i>Zeroes and dashes, anonymous</i>

Back

Import

Connect a source

After you configure Identity Resolution settings, the next step is to connect a [source](#) to the Segment space.

Create an audience

After you connect a source, Segment creates user profiles based off of replayed and newly incoming data.

Welcome Sloth, let's get your new space up and running!



Set Your Identity Rules

First, set identity rules to define the data content of user profiles.

[View Identity Settings](#)



Connect Your Sources

Select which upstream sources you would like to receive data to build your audiences, traits, and profiles.



Build Your First Audience

Create a cohort of users based on your desired parameters.

[Create Audience](#)

The next step, which is important in the Dev space, is to create an audience to ensure that user profiles have populated correctly and that the Identity Resolution settings follow expected business logic.

For example, if there should be 100,000 distinct users who have a `user_id`, this would be a great way to validate that the Identity Resolution settings have calculated profiles correctly.

For more information about how to create audiences and traits, see Segment's [Audiences docs](#).

This page was last modified: 15 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](#)

© 2025 Segment.io, Inc.

[Privacy](#)

[Terms](#)

[Website Data Collection Preferences](#)

