⊘ Mobile	⊘ Mobile
⊘ Server	⊘ Server

Our Twitter Ads destination code is all open-source on GitHub if you want to check it out.

Getting Started

From your Segment Destinations page click on Twitter Ads. Twitter Ads can track conversions in two ways. You can opt to use their newest **Universal Website Tag** or their **Single Event Website Tag**. You could also use both types of tags on the same page.

If you want to use the universal tag, all you have to do is enter your **Universal Website Tag Pixel ID** in the settings and create your conversion events inside Twitter's UI. The **Universal Website Tag** will map to any Segment event listed in the table Advanced Conversion Tracking/Dynamic Ads. You can find this pixel ID in Twitter's snippet that is generated for you after making a new Universal Conversion Event:

How to setup your universal website tag

Install your website tag

If you are the person who will be installing the website tag, download the code snippet and place it at the bottom of all pages on your website before the </body> HTML tag. Visit the website tag help and troubleshooting page for help.

⬇ **Download a txt file with this code snippet**

```
<!-- Twitter universal website tag code -->
<script>
!function(e,n,u,a){e.twq||(a=e.twq=function(){a.exe?a.exe.apply(a,arguments):
a.queue.push(arguments);},a.version='1',a.queue=[],t=n.createElement(u),
t.async=!0,t.src='//static.ads-twitter.com/uwt.js',s=n.getElementsByTagName(u)[0],
s.parentNode.insertBefore(t,s))}(window,document,'script');
// Insert Twitter Pixel ID and Standard Event data below
twq('init','XXXXXX');
twq('track','PageView');
</script>
<!-- End Twitter universal website tag code -->
```

If you'd like to use the **Single Event Website Tags**, you must map the Segment events to your tags' respective conversion pixel IDs. The **Single Event Website Tag** is good to use if your event names do not match the Segment supported events with the **Universal Website Tag**.

Note that Twitter Ads only works with Segment's client-side JavaScript library `analytics.js` as it requires its script or pixel to be loaded onto your website.

Page

If you are using the **Universal Website Tag**, when you call `.page()` we will fire a standard pageview event using Twitter's JS library: `window.twq('track', 'PageView')` as seen in their auto generated snippet.

If you are only using **Single Event Website Tags**, we will load their standard pageview pixel with default query values (`revenue: 0`, `quantity: 0`) given that you have provided the **Page Pixel** in your Segment settings for Twitter Ads.

If you have both tags defined in your destination settings, we will load **both** the single event pixel as well as the use the universal JS library to send the pageview event.

Track

Single Event Website Tags

Each time you `track` an event from the browser with our Twitter Ads destination enabled we compare the event name to the events you have mapped to **Twitter Conversion IDs** inside of Segment. If the event being tracked corresponds to a Twitter conversion, a Twitter conversion pixel will be loaded on that page with the ID that you mapped it to.

You can find your conversion pixel IDs after you create the single conversion event inside Twitter:

order completed Purchase Website Tag ID XXXX	Single event website tag	UNVERIFIED ❓ View Code	30 days	1 day	ENABLED Audience Manager	Edit

For example, if your mapping in Segment looks like this:

Events

Twitter recognizes pixel ids, not custom events. When you `analytics.track(event, properties)` an event that represents a Twitter conversion, you'll need to map the event name on the left to it's corresponding Twitter pixel id on the right. Contact us if you need help!

| Signed up | ▸ | 28j3fo | ⊗ |

⤇ Add row

Then every time you make this `track` call (from the browser):

```
analytics.track('Signed Up', {
  quantity: 1,
  revenue: 10
});
```

We will fire a Twitter conversion pixel for you with a **Twitter Conversion ID** of `28j3fo`.

It doesn't matter what you call the conversion inside of Twitter Ads as long as you paste your **Twitter Conversion ID** next to the right event in Segment.

If you include `properties.quantity` and `properties.revenue` with your events, we will send that along as parameters of the conversion tag. If these two properties are not provided, we will send the two parameters as `0`.

Thus the example `.track()` event above would fire this tag:

```
<img src="http://analytics.twitter.com/i/adsct?txn_id=28j3fo&p_id=Twitter&tw_sale_amount=10&tw_order_quantity=1">
```

Advanced Conversion Tracking and Dynamic Ads

Our Twitter destination supports Advanced Conversion Tracking which can be used to run **Dynamic Ads Campaigns**. Once you create your conversion events using **Universal Website Tag** inside Twitter's UI, just follow our Ecommerce V2 Spec and we will automatically map select events (listed below). Keep in mind that you **must** provide the **Universal Website Pixel ID** in your settings for this feature to work.

NOTE: For each of these Segment events, we will still check if you had mapped any of them to a **Single Event Website Tag** *in your Segment settings and fire those tags in addition to these advanced conversion tags.*

The following table shows which Segment events map to which type of Twitter tags:

Segment Event Name	Twitter Tag Type
Order Completed	Purchase
Product Viewed	ViewContent
Product Added	AddToCart
Products Searched	Search
Product Added To Wishlist	AddToWishlist
Checkout Started	InitiateCheckout
Payment Info Entered	AddPaymentInfo

The following table show how the properties of Segment events would map to Twitter's supported parameters:

Segment Property Name	Twitter Parameter
value / revenue / (price * quantity)	value
currency	currency
products.$.product_id / products.$.sku	content_ids
N/A	content_type
products.$.name	content_name
products.$.category	content_category
products.$.quantity	num_items
order_id	order_id
status	status

IMPORTANT:

- You can choose in your Segment settings if you want to populate `content_ids` parameter with your `product_id`(s) or `sku`(s). Whatever you choose should match the IDs in your Product Feed if you are running dynamic ads.

- `content_type` will always be correctly hardcoded to be `'product'`.

- `content_name` and `content_category` for multiple products will be concatenated delimited by commas.

- `num_items` represents the sum total of all your products' respective quantities.

- While `properties.status` is not spec'd by Segment, you can still send that property through and we will map it to Twitter's `status` parameter, which is an optional free text field representing the state of the conversion event (eg. 'completed', 'in review', 'processed', etc.)

- `value`, `currency`, `order_id` and `num_items` will not be mapped for any pre-purchase tags because it will attribute revenue, which is undesired behavior for ecommerce/retail businesses.

The following code snippets represent the code we would fire under the hood on your webpage given an example Segment event.

> ℹ️
> The following assumes that the setting for* **Product Identifier** *is `product ID` (it can also be SKU).

Order Completed -> Purchase

```
analytics.track('Order Completed', {
  orderId: '50314b8e9bcf000000000000',
  total: 30,
  revenue: 25,
  shipping: 3,
  tax: 2,
  discount: 2.5,
  coupon: 'hasbros',
  currency: 'USD',
  repeat: true,
  products: [
    {
      product_id: '507f1f77bcf86cd799439011',
      sku: '45790-32',
      name: 'Monopoly: 3rd Edition',
      price: 19,
      quantity: 1,
      category: 'Games'
    },
    {
      product_id: '505bd76785ebb509fc183733',
      sku: '46493-32',
      name: 'Uno Card Game',
      price: 3,
      quantity: 2,
      category: 'Games'
    }
  ],
  status: 'completed!'
});
```

```
window.twq('track', 'Purchase', {
  value: '25.00', // Required
  currency: 'USD', // Required. Defaults to 'USD'
  content_ids: ['505bd76785ebb509fc183733', '507f1f77bcf86cd799439011'], // Required
  content_type: 'product', // Required
  content_name: 'Monopoly: 3rd Edition, Uno Card Game',
  num_items: '3',
  order_id: '50314b8e9bcf000000000000',
  status: 'completed!'
});
```

Product Viewed -> ViewContent

```
analytics.track('Product Viewed', {
  product_id: '507f1f77bcf86cd799439011',
  sku: 'G37',
  category: 'eSports',
  name: 'Starcraft: Brood War',
  brand: 'Blizzard',
  variant: 'Expansion Pack',
  price: 12.99,
  quantity: 2,
  coupon: 'STOPTHETILT',
  currency: 'USD',
  position: 1,
  value: 0.50
});
```

```
window.twq('track', 'ViewContent', {
  content_ids: ['507f1f77bcf86cd799439011'], // Required
  content_type: 'product', // Required
  content_name: 'Starcraft: Brood War',
  content_category: 'eSports'
});
```

NOTE: *For this event, since `properties.value` is spec'd by Segment. we will respect that property over the `properties.price * properties.quantity` since the former can be a different explicit monetary value.*

Product Added -> AddToCart

```
analytics.track('Product Added', {
  cart_id: 'yolo123',
  product_id: 'kenbone777',
  sku: '1',
  category: 'Political Action Figure',
  name: 'The Bone Zone',
  brand: 'Winner',
  variant: 'Original',
  price: 1738.00,
  quantity: 1,
  coupon: 'KENBONE',
  position: 1
});
```

```
window.twq('track', 'AddToCart', {
  content_ids: ['kenbone777'], // Required
  content_type: 'product', // Required
  content_name: 'The Bone Zone'
});
```

Products Searched -> Search

```
analytics.track('Products Searched', { query: 'gluten rich food' });
```

```
window.twq('track', 'Search', {});
```

NOTE: Since no parameters are required for this tag and its supported ones do not correspond with this spec'd event's properties, we will not map any properties for this tag other than `properties.status`.

Product Added To Wishlist -> AddToWishlist

```
analytics.track('Product Added to Wishlist', {
  wishlist_id: 'cmon-santa-bro',
  wishlist_name: 'This is the year',
  product_id: '507f1f77bcf86cd799439011',
  sku: 'G-32',
  category: 'TV Shows',
  name: 'Rick and Morty: Season 3',
  brand: 'Adult Swim',
  variant: 'DVD',
  price: 19.99,
  quantity: 1,
  coupon: 'RICKPOTION',
  position: 1
});
```

```
window.twq('track', 'AddToWishlist', {
  content_name: 'Rick and Morty: Season 3',
  content_category: 'TV Shows',
  content_ids: ['507f1f77bcf86cd799439011']
});
```

Checkout Started -> InitiateCheckout
```

```
analytics.track('Checkout Started', {
  order_id: '50314b8e9bcf000000000000',
  affiliation: 'Google Store',
  revenue: 25,
  shipping: 3,
  tax: 2,
  discount: 2.5,
  coupon: 'hasbros',
  currency: 'USD',
  products: [
    {
      product_id: '507f1f77bcf86cd799439011',
      sku: '45790-32',
      name: 'Monopoly: 3rd Edition',
      price: 19,
      quantity: 1,
      category: 'Games'
    },
    {
      product_id: '505bd76785ebb509fc183733',
      sku: '46493-32',
      name: 'Uno Card Game',
      price: 3,
      quantity: 2,
      category: 'Games'
    }
  ]
});
```

```
window.twq('track', 'InitiateCheckout', {
  content_ids: ['505bd76785ebb509fc183733', '507f1f77bcf86cd799439011'],
  content_name: 'Monopoly: 3rd Edition, Uno Card Game',
  content_category: 'Games, Games'
});
```

Payment Info Entered -> **AddPaymentInfo**

```
analytics.track('Payment Info Entered', {
  checkout_id: 'co-120387',
  order_id: '182hjdld9sl',
  step: 4,
  shipping_method: 'FedEx',
  payment_method: 'Credit Card'
});
```

```
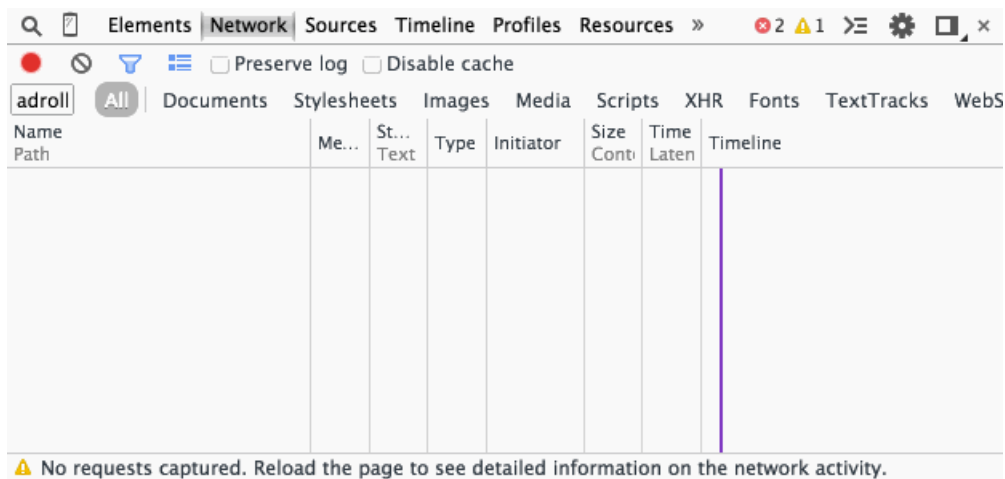window.twq('track', 'AddPaymentInfo', {});
```

NOTE: Since no parameters are required for this tag and its supported ones do not correspond with this spec'd event's properties, we will not map any properties for this tag.

Troubleshooting

Script unverified or undetected by third-party tool

Many times this is a limitation on the tool's detection process, where the detector is looking for a specific HTML element on your page. Our client side analytics.js library asynchronously loads the tool's library or pixel onto the page. As such, the detection fails.

In order to confirm that the tool's library or pixel is actually loaded onto the page, you can open up the JavaScript console and go to the network tab when the page is loading.

If the script isn't loading, check that any form of ad blocker is disabled.

Settings

Segment lets you change these destination settings from the Segment app without having to touch any code.

SETTING	DESCRIPTION
Events	`text-map`, defaults to {}. Twitter recognizes pixel ids, not custom events. When you `analytics.track(event, properties)` an event that represents a Twitter conversion, you'll need to map the event name on the left to it's corresponding Twitter pixel id on the right. Contact us if you need help!
Product Identifier	`select`, defaults to productId. If you are using Advanced Conversion Tracking, you can choose to send Product IDs or SKUs as the content_ids. Whatever you choose should match the IDs in your Product Feed if you are running dynamic ads.
Page Pixel	`string`. Twitter recognizes pixel ids. This is just a special pixel id you can use for page views. It will get executed on any call to `analytics.page()`.
Universal Website Tag Pixel ID	`string`. Twitter supports having a single universal website tag for conversions. Please enter your Universal Website Tag Pixel ID. This setting is also required in order to use Advanced Conversion Tracking.

This page was last modified: 13 Apr 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