

July 17, 2015 Rev 2.4

Peter Gustafson

Target link: <https://stripe.com/docs/orders/tax-integration>

## Automated Tax Calculations

Use a tax provider to automatically calculate the taxes due on your Stripe orders.

Charging the legally appropriate amount of tax on orders is especially tricky for online sales. The right percentage to charge—if any—depends upon the customer's country or US state, the types of products being purchased, and the order total. To help businesses dynamically calculate and apply accurate taxes in real-time, Stripe has partnered with Avalara, TaxJar, and Taxamo.

We've integrated these tax providers' tax calculation product directly into the Stripe Orders API. The Orders API allows users to represent their inventory and orders within Stripe, rather than on their own site or through a third-party. If you are not yet familiar with the Orders API, we recommend that you first visit the Orders API Overview documentation before diving into tax calculations.

To enable dynamic tax calculations for your orders, follow our guides to get started with Avalara, TaxJar, or Taxamo. You can optionally read on below to learn how the integration works. If you sell a range of products with different tax implications, see how to use tax codes. If you also sell directly to businesses and need to factor in value-added tax (VAT), Stripe supports that, too.

### How the integration works

The Orders API tax provider integrations automatically manage everything necessary to compute and commit tax quotes for your orders. To help explain how the integration functions, read the high-level and technical overviews below. (The latter is intended for developers.)

### High-level overview

When you create an order using the Orders API, Stripe programmatically sends the details to your preferred tax provider:

The specific versions (as SKUs) of the products being ordered

Your origination shipping address

The customer's destination shipping address

Orders API tax providers need all three components to calculate taxes for your orders. It is therefore important that you either pass shipping details directly or attach a customer with a valid shipping address when the order is created. The provider determines the appropriate amount of taxes using the order information provided and returns this information to Stripe, which we add to the order in real-time. Your customers pay the updated total, and you can later

view the charged tax amount in your Dashboard and in your tax provider's reports and dashboard.

The tax estimate service is usually provided free of charge by the tax providers, so creating orders with Stripe does not by itself cost you anything. Once the order is paid for, Stripe commits the estimate with the provider. Tax providers charge based on committed estimates in most cases, but you should check with the provider you are using.

Should an order later be returned or canceled, Stripe automatically voids the transactions with the provider to help you keep your tax reports accurate. The integration works seamlessly both when an order is fully or partially returned.

#### Technical overview

Looking at the process from a more technical perspective, once you set up your tax provider integration, you should automatically see tax items in your order's items array. For example, the following request creates an order shippable to a US address:

#### Terminal

```
curl https://api.stripe.com/v1/orders \
  -u sk_test_qMgXKP811TZDUCSKV7PN5YZQ: \
  -d items[][parent]=tax_provider_test_sku \
  -d currency=usd \
  -d shipping[name]="Jenny Rosen" \
  -d shipping[address][line1]="123 Main Street" \
  -d shipping[address][postal_code]=94110 \
  -d shipping[address][city]="San Francisco" \
  -d shipping[address][state]=CA \
  -d shipping[address][country]=US \
  -d email="jenny@example.net"
```

When processing the request, Stripe uses the tax provider's API to obtain tax line items and includes them directly in the response. For this example, the response below includes the 7.5% California sales tax applicable to the order (some fields removed for brevity):

```
{
  "id": "or_17pkOFDAu10Yox5ReR89xhT1",
  "object": "order",
  "amount": 7524,
  ...
  "items": [
    {
      "object": "order_item",
      "amount": 6999,
      "currency": "usd",
```

```

    "description": "Slim Jeans",
    "parent": "tax_provider_test_sku",
    "quantity": 1,
    "type": "sku"
  },
  {
    "object": "order_item",
    "amount": 525,
    "currency": "usd",
    "description": "CA STATE TAX (PC040100)",
    "parent": null,
    "quantity": null,
    "type": "tax"
  }
],
...
"shipping": {
  "address": {
    "city": "San Francisco",
    "country": "US",
    "line1": "123 Main Street",
    "line2": null,
    "postal_code": "94110",
    "state": "CA"
  },
  "name": "Jenny Rosen",
  "phone": null
},
...
}

```

### Setting up product tax codes

Some jurisdictions apply different tax policies to different product types, such as food and clothing. With Stripe's tax provider integration, appropriate taxes can be calculated on your products that have exclusions or varying tax rates.

To start, assign a tax code to your products stored in Stripe by setting the `tax_code` metadata value on the product objects. The value associated with the metadata field `tax_code` will be passed directly to the tax provider.

The `tax_code` metadata property can be set either in the Dashboard or using the API. To use the Dashboard, navigate to a product page, click the Edit button under Metadata and add a value for `tax_code`.

To use the API, perform an update product request like the one below:

Terminal

```
curl https://api.stripe.com/v1/products/pr_6rpcQjKz7PexY2 \  
-u sk_test_qMgXKP811TZDUCSKV7PN5YZQ: \  
-d metadata[tax_code]=AB230
```

Including VAT

The Orders API also supports VAT for business-to-business transactions. Merely set a VAT ID on the Stripe Customer object and then use that customer for an order. This request adds a VAT ID to an existing customer:

Terminal

```
curl https://api.stripe.com/v1/customers/cus_XSaJPxdArYmWuWCI \  
-u sk_test_qMgXKP811TZDUCSKV7PN5YZQ: \  
-d business_vat_id=DK123456
```

When that customer is subsequently used for an order, the tax provider will receive the VAT ID and calculate the appropriate tax amount.

Questions?

We're always happy to help with code or other questions you might have! Search our documentation, contact support, or connect with our sales team. You can also chat live with other developers in #stripe on freenode.