

AvaTax Integration

Storefront First Architecture

Version 4.0



Table of Contents

1. Summary	3
2. Component Overview	4
2.1 Functional Overview	4
2.2 Use Cases	4
2.3 Limitations, Constraints.....	4
2.4 Compatibility.....	4
2.5 Privacy, Payment.....	4
3. Implementation Guide.....	5
3.1 Setup	5
3.1.1 Import Cartridges	5
3.1.2 Import Metadata	5
3.1.3 AvaTax Reconciliation Utility Configuration	5
3.2 Configuration	6
3.2.1 Configuration for AvaTax Services	6
3.2.2 AvaTax Settings Configuration	7
3.3 Services	12
3.4 Testing	12
3.5 Error Handling	13
3.6 External Interfaces.....	13
4. Operations, Maintenance	13
4.1 Data Storage.....	13
4.2 Availability.....	13
4.3 Support	14
5. Known Issues	14
6. Release History	15

1. Summary

Avalara is a leading provider of webservice-based sales tax automation solutions (AvaTax) for businesses of all sizes. As the leading web-hosted on-demand sales tax solution, AvaTax is offered as a Software-as-a-Service (SaaS) alternative to more expensive and less adaptive software sales tax calculating and management solutions. AvaTax integrates seamlessly with the market leading financial applications to provide end-to-end compliance that is completely automated.

The AvaTax SFRA cartridge provides rapid integration for SFRA implementations Salesforce B2C Commerce. The AvaTax cartridge is a self-contained package that can easily integrate into any project. This cartridge can be configured in the Business Manager and contains all elements necessary to perform a successful implementation of AvaTax.

2. Component Overview

2.1 Functional Overview

1. AvaTax OnDemand is a powerful, online sales tax compliance solution that provides all businesses with a level of sales tax automation that until now has been available only to Fortune 500 companies. AvaTax OnDemand's transaction-based service model and seamless integration into business applications gives you rapid access to all jurisdiction assignments and real-time sales tax calculations. Cutting-edge technologies and superior processing logic can manage even the most complicated tax issues, such as situs, nexus, tax tiers, tax holidays, exemptions, certificate management and product taxability rules.
2. During checkout, the shopping cart and customer information is transmitted via the AvaTax API web service specifications. Tax information is sent back and applied to products in the cart.
3. Advanced Address Validation logic allows to enable/disable the AVS calls for US and Canada countries.
4. Detailed transaction logging on the Admin console.

2.2 Use Cases

This cartridge can be used to calculate taxes on the SFRA storefront through the AvaTax REST v2 API web service. It can also be used for address validation for all the US and Canada addresses.

Following methods are provided as part of the cartridge:

1. Calculate Tax
2. Validate Address
3. Commit Transaction
4. Cancel Transaction

2.3 Limitations, Constraints

As of now, only limitation is that AvaTax address validation is only available for US and Canada.

2.4 Compatibility

This release is compatible with the latest SFRA version 4.2.1.

2.5 Privacy, Payment

Customer purchase information for example - ship-to-address, items in the cart, their quantity, amount, discounts etc. are sent to and from AvaTax for tax calculations and address verification. No customer payment information is accessed.

3. Implementation Guide

3.1 Setup

3.1.1 Import Cartridges

Import the *bm_avatax*, *int_avatax_sfra* and *int_avatax_svcclient* into the UX Studio workspace

1. Open UXStudio in Eclipse IDE.
2. Click **File > Import > General > Existing projects** into workspace.
3. Browse to the directory where the cartridges 'bm_avatax', 'int_avatax_sfra' and 'int_avatax_svcclient' have been downloaded.
4. Click **Finish**.
5. Click **OK**, when prompted to link the cartridges to the sandbox.
6. Modify the site path in **Business Manager > Administration > Manage Sites**. Make sure the cartridge name 'int_avatax_sfra' appears before 'app_storefront_base'.

3.1.2 Import Metadata

Import the site-import metadata into the Business Manager.

1. Log in to the Business Manager.
2. Click **Administration > Site Development > Site Import & Export**.
3. Use the upload control to browse for the site-import.zip file located in the **metadata** folder.
4. Click **Upload**.
5. Select the site-import.zip and click **Import**. Import should complete successfully.

3.1.3 AvaTax Reconciliation Utility Configuration

1. Make sure the 'bm_avatax' cartridge is imported in UX Studio workspace.
2. Add bm_avatax cartridge in BM cartridge path. **Administration > Sites > Manage Sites > Business Manager - Settings**
3. Make sure the AvaTax BM Module has been added to Business Manager Modules and has been assigned necessary permissions as follows.
 - Go to **Administration > Organization > Roles & Permissions**
 - Select the Role to give access to. e.g. Administrator
 - Go to **Business Manager Modules** section. Select the context for the Site or Organization and click **Apply**.
 - In the list of BM modules that are displayed, check the checkbox against **AvaTax**. And, click **Update**.
 - A new module AvaTax and its menu items should appear under Merchant Tools.

3.2 Configuration

3.2.1 Configuration for AvaTax Services

Make sure the AvaTax services have imported as part of site import step (3.1.2) performed above. Below service configurations should be present in the Services section.

1. Log into the Business Manager.
2. Navigate to - **Administration > Operations > Services**
3. Configure your AvaTax credentials under **Credentials** tab, in **credentials.avatax.rest**.
4. Make sure the service URL used is one of these:
 - Development : <https://sandbox-rest.avatax.com>
 - Production : <https://rest.avatax.com>

Notes:

- A service named **logentries.avatax.svc** should also have been created with corresponding Profile and Credential item. Please **DO NOT** modify/remove these.
- Do not modify service name(s) of any of the services.

Below are the screenshots for reference.

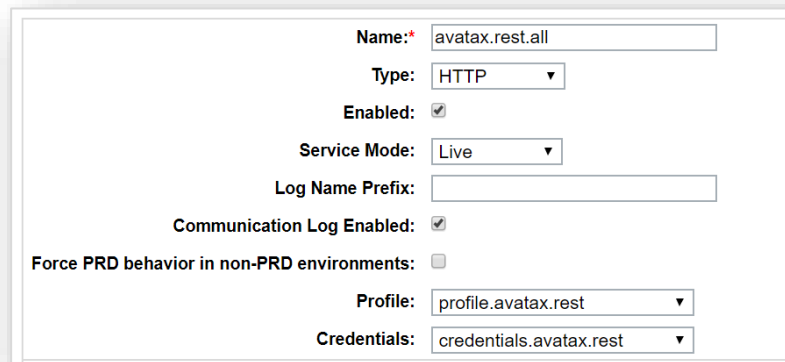
Credential

Name:*	credentials.avatax.rest
URL:	https://sandbox-rest.avatax.com/
User:	XXXXXXXXXX
Password:	*****

Profile

Name:*	profile.avatax.rest
Connection Timeout (ms):	20,000
Enable Circuit Breaker:	<input type="checkbox"/>
Max Circuit Breaker Calls:	0
Circuit Breaker Interval (ms):	0
Enable Rate Limit:	<input type="checkbox"/>
Max Rate Limit Calls:	0
Rate Limit Interval (ms):	0

Service



The image shows a configuration dialog box for a service. The fields are as follows:

- Name: avatax.rest.all
- Type: HTTP
- Enabled:
- Service Mode: Live
- Log Name Prefix: (empty)
- Communication Log Enabled:
- Force PRD behavior in non-PRD environments:
- Profile: profile.avatax.rest
- Credentials: credentials.avatax.rest

3.2.2 *AvaTax Settings Configuration*

Configure AvaTax settings using the AvaTax module in the Business Manager.

1. Log into the Business Manager.
2. Select the desired site from the dropdown.
3. Navigate to **Merchant Tools > AvaTax > AvaTax Settings**
4. Select all applicable settings for the storefront. Below is the description for these settings in brief.

AvaTax Options

- **Tax Calculation** - When selected, this will enable AvaTax tax calculation on transactions.
- **Address Validation** - When selected, this will enable shipping address validation during checkout on the storefront. AvaTax only validates addresses in the United States and Canada.
- **Save Transactions to AvaTax** - When selected, this will save all orders to AvaTax. When set to 'Disabled', orders will not be saved to AvaTax, and so features such as voiding, modifying the transactions will not be available.
- **Commit Transactions to AvaTax** - If enabled, successful orders on the storefront will be marked 'Committed' when posting to AvaTax.
- **Company Code** - AvaTax company code configured on the admin console of the AvaTax account.
- **Customer code** - Select what value is sent as the customer code for authenticated customers. For guest customers, Email ID entered during checkout is used.
- **Shipping Tax Code** - This tax code will be used to calculate taxes for shipping methods. If not specified, freight code 'FR' will be used.

Address Details

- **Location Code** - If the Location Code value is provided, AvaTax prefers it over other address fields for tax calculation.
- **ShipFrom Address fields** - Details of the Company location from where the products will be shipped. This will be used for calculating taxes for all transactions.

Address line 1, line 2, line 3, City, State, Zip/Postal Code, and Country Code should be filled out as per exact Company address to help calculate taxes accurately.

Note:

- Navigate to **Merchant Tools > Site Preferences > Order > Order Access Settings**, and set Limit Storefront Order Access to **No**.
- Make sure to select the required Promotion Preference by Navigating to **Merchant Tools > Site Preferences > Promotion > Discount Taxation**.

3.3.1 VAT and Invoice Messages for EU countries

This section is important if the storefront operates in the European (EU) countries. It can be skipped if the storefront doesn't ship any products to EU countries.

AvaTax service returns taxes and invoice messages based on shipTo and shipFrom location in European countries.

These messages are stored as a custom attribute in -

1. *Basket* object, if the order is still not placed
2. *Order* object, after the order has been placed

The message object is stored in below JSON format.

```
{"InvoiceMessageMasterList":[{"MessageCode":0,"Message":"No applicable messaging for this line."}, {"MessageCode":1,"Message":"Intra-EU Supply of Goods as per Art. 138 EU VAT Directive 2006/112"}], "InvoiceMessageList":[{"TaxLineNo":"98ed5002d09d3b33abba8b10ed", "MessageCode":1}, {"TaxLineNo":"f9a79cafa05441b14e72c0e2a6", "MessageCode":0}]}
```

The messages may be utilized to display to the customers during checkout as required.

Description:

- 'InvoiceMessageMasterList' is an array of all applicable messages.
- 'MessageCode' is the index used to refer to the corresponding 'Message'.
- 'InvoiceMessageList' is an array of all line items in the Basket and their corresponding MessageCodes.
- 'TaxLineNo' indicates UUID of the line items in the current Basket object.

For example, - {"TaxLineNo":"98ed5002d09d3b33abba8b10ed", "MessageCode":1}
This indicates that a line item in the current basket with UUID
98ed5002d09d3b33abba8b10ed has a VAT invoice message 'Intra-EU Supply of

Goods as per Art. 138 EU VAT Directive 2006/112' which corresponds to MessageCode 1.

3.3.2 Customer code

Customer code sent to Avalara services can be customized in the AvaTax Settings. By default Email ID provided during the checkout process is sent over to AvaTax. If the customer is authenticated, and the custom attribute preference value is available, the attribute value from 'Profile' object is used.

The order of preference used for customer code is -

- Custom field attribute provided in the custom preferences section (Authenticated users and subject to availability)
- Email ID provided during checkout (For both authenticated and guest users)
- Email ID from the User Account (Authenticated users only)

3.3.3 Tax Breakdown details

When the taxes are calculated and returned by Avalara services, AvaTax cartridge stores the tax details for every line item (Product, Shipping, Gift certificate etc.) in the Basket or Order object in the form of a custom attribute based on whether the order is in placed state or not. These details can either be used to display to the end user or for reporting purposes.

- If the *Basket* is still in checkout process, i.e. If the order is not yet placed, the tax details are stored in a custom attribute of *Basket* object.

The attribute name is - `ATTaxDetail`. It can be accessed as -
`basket.custom.ATTaxDetail`

- If the order is in placed state, the tax details are stored on Order system object.

The attribute name is - `ATTaxDetail`. It can be accessed as -
`order.custom.ATTaxDetail`

The tax details are stored in the form of JSON string. It can be utilized as required.

Below is an example of tax details JSON for a *Basket*.

```
[
{
  "lineitemid":"750518703299",
  "shipmentid":"me",
  "taxes":[
    {
      "jurisdictiontype":"State",
      "jurisdiction":"WASHINGTON",
      "exempt":0,
      "nontaxable":0,
```

```
"taxable":299.99,  
"rate":0.065,  
"tax":19.5
```

```

    },
    {
      "jurisdictiontype": "County",
      "jurisdiction": "KITSAP",
      "exempt": 0,
      "nontaxable": 0,
      "taxable": 299.99,
      "rate": 0,
      "tax": 0
    },
    {
      "jurisdictiontype": "City",
      "jurisdiction": "BAINBRIDGE ISLAND",
      "exempt": 0,
      "nontaxable": 0,
      "taxable": 299.99,
      "rate": 0.025,
      "tax": 7.5
    }
  ]
},
{
  "lineitemid": "me",
  "shipmentid": "me",
  "taxes": [
    {
      "jurisdictiontype": "State",
      "jurisdiction": "WASHINGTON",
      "exempt": 0,
      "nontaxable": 0,
      "taxable": 9.99,
      "rate": 0.065,
      "tax": 0.65
    },
    {
      "jurisdictiontype": "County",
      "jurisdiction": "KITSAP",
      "exempt": 0,
      "nontaxable": 0,
      "taxable": 9.99,
      "rate": 0,
      "tax": 0
    },
    {
      "jurisdictiontype": "City",
      "jurisdiction": "BAINBRIDGE ISLAND",
      "exempt": 0,
      "nontaxable": 0,
      "taxable": 9.99,
      "rate": 0.025,
      "tax": 0.25
    }
  ]
}
]

```

JSON attributes:

1. `lineitemid` represents the object ID of the line item from Business Manager. For example, for the product line item, this value would be its Product ID.
2. `shipmentid` attribute in each object represents the Shipment to which current line item belongs.
3. `jurisdictiontype` represents the tax jurisdiction of in which the current tax amount is calculated. e.g. Country, State, City, County, Special etc.
4. `jurisdiction` represents the jurisdiction name. e.g. Canada, Ontario, California etc.

5. `taxname` represents a short tax description for the tax applied. e.g. CANADA GST/TPS, TN STATE TAX etc.
6. `nontaxable` has a value which is not taxable `taxable` attribute indicates the amount that is taxed.
7. `exempt` attribute indicates the amount which is tax-exempt.
8. `rate` contains the tax rate used for calculating the taxes `tax` has a value which is the tax that is calculated.

3.3 Services

Some features are also included in this integration as experimental ones. These can be accessed in the AvaTax Settings menu under AvaTax.

Navigate to - **Merchant Tools > AvaTax > AvaTax settings > Services**

- **Test Connection** - This can be used to verify whether connection to AvaTax is authenticated/successful. If it indicates that the connection is not authenticated, you may want to have a look at the Service credentials configured in step 3.2.1
- **Void Transaction** - This feature can be used void/cancel an order posted earlier to AvaTax. It cancels the order, or it does not, depending on if the order can be cancelled or not. In either case the service response is returned and displayed in the text area below this section.
- **Commit Transaction** - This feature can be used commit an order posted earlier to AvaTax. It marks the order committed or it does not, depending on if the order can be committed or not. In either case the service response is returned and displayed in the text area below this section.
- **Validate Address** - This feature can be used to validate shipping address of an order and receive a validated address as response.

All the above features can be scaled to process many orders on the storefront.

3.4 Testing

Running unit tests

You can run `npm test` in the terminal or windows command prompt to execute all unit tests in the project. Unit tests are present under `/test/unit` directory.

Running integration tests

Integration tests are located in the `/test/integration` directory.

To run integration tests you can use the following command:

```
npm run test:integration
```

Note: Please note that short form of this command will try to locate URL of your sandbox by reading dw.json file in the root directory of your project. If you don't have dw.json file, integration tests will fail. sample dw.json file:

```
{
  "hostname": "your-sandbox-hostname.demandware.net",
  "username": "yourlogin",
  "password": "yourpwd",
  "code-version": "version_to_upload_to"
}
```

You can also supply URL of the sandbox on the command line:

```
npm run test:integration -- --baseUrl devxx-sitegenesis-dw.demandware.net
```

3.5 Error Handling

This cartridge works as plugin hence all the exception and error are handled internally. Plugin displays address validation error message on the cart and my account page, if address validation is enabled. For tax calculation on the cart, in case of errors, cartridge will not stop a customer from placing the order. The errors are logged in the custom log files on the Business Manager.

3.6 External Interfaces

All requests are done through AvaTax REST v2 API. The full reference guide, along with the resource structure for requests, can be found on their developer site -

<https://developer.avalara.com/api-reference/AvaTax/rest/v2/>

4. Operations, Maintenance

4.1 Data Storage

The only custom attributes stored in B2C are the ones created during the initial import. These are described in the [Configuration section](#).

4.2 Availability

Cartridge functionality will be dependent on the availability of the AvaTax API service. Current AvaTax operational status can be viewed here - <https://status.avalara.com/>

4.3 Support

If you encounter any issues related to configuring or testing this integration, you may contact Avalara customer support for assistance. You can submit a case by email, online web-to-case, telephone or chat. Please go to <http://avalara.com/Technical-Support> for more details as availability will vary depending on your support package.

5. Known Issues

No known existing issues. Please report any issues to support@avalara.com.

6. Release History

Version	Date	Changes
1.0	19 April 2018	Initial release
2.0	25 th May 2018	Introduced below items: <ol style="list-style-type: none">1. A method to validate AvaTax credentials.2. A configuration to allow users to choose if they want to save their transactions at AvaTax.
3.0	07 th Dec 2018	Added support for VAT and Invoice Messages for EU countries. Reconciliation Utility BM Extension Commit Transaction on successfully placing order Custom Customer Code support
4.0	9 th Aug 2019	Added - <ul style="list-style-type: none">- Experimental features section- AvaTax Settings section