

## Avalara Tax Integration

ProfitKey's Avalara Tax connector automatically exchanges data between PKI's ERP solution and Avalara's Sales Tax software. Avalara's industry leading Sales Tax software helps businesses manage all types of taxes including: sales, use, excise, GST, VAT, and other tax types for USA and Foreign tax jurisdictions.



PKI's Avalara Tax connector saves staff time, eliminates redundant data entry and eliminates costly data entry errors.

With the PKI Avalara Tax connector:

- A ProfitKey user can segregate products into the appropriate tax categories one time; so that ensuing orders that might have multiple products, with multiple different tax treatments, from multiple taxing authorities, get all their tax amounts calculated correctly and inserted into the order without leaving the PK ERP solution.
- A ProfitKey users does not need to leave the PK Solution, re-enter every line item for the order into Avalara's solution, and then copy and enter those amounts into the line items in the PK ERP Solution. The PKI Avalara Tax Connector helps save substantial staff time and eliminates any chance for entry errors while bringing the tax amounts into PKI ERP.

PK ERP Release 8.1.5, or higher, is required for using the connector.

The connector provides integration for all of Avalara's Required list of components and has received Avalara's Certification Badges for [Sales Tax](#) and [Address Validation](#). The components provide integration across five areas:

1. Administration & Utilities
2. Customer Record Integration
3. Items/Charge Integration
4. Sales/Billing Documentation Integration
5. Server Audit Clarity and Installation Requirements

### 1. ADMINISTRATION & UTILITIES:

#### AvaTax configuration window

The AvaTax Configuration Dialog Window must allow the user to specify the configuration/connection information

- Account Number
- License Key
- Service URL
- Company Code Lookup

#### AvaTax Test Connect Button

Test the connection to the AvaTax service and verify the AvaTax credentials. This is an important element to allow for successful troubleshooting of the AvaTax Service.

#### Control - Disable Document Recording

In order for this connector to be used in conjunction with other integrations to AvaTax, the user must be able to control which connector is used for recording documents to AvaTax. From a technical standpoint, simply use DocType: 'SalesOrder' on all calls and suppress any non-getTax calls (i.e. cancelTax, postTax).

Note: Other documentation may refer to Document Committing, this is interchangeable with Document Recording

#### **Disable AvaTax Option**

The user must have an option to turn on or off the AvaTax Calculation service independent of any other Avalara product or service.

#### **Enable client-side logging**

Enables detailed AvaTax transaction logging within the application including capture of round-trip processing time. We need the complete request/response for each call made to Avalara services. It does not need to run all the time as we understand the database will grow unnecessarily large – you are free to only log the last week, 30 days, custom, or have a control for the next N hours, etc. The spirit of the requirement is to assist customers and support in troubleshooting exercises, so it needs to be retrievable by an end user (or an administrator). It should be specifically Avalara service calls.

#### **User Implementation Guide**

The User Implementation Guide should contain screenshots and information allowing the end user to configure for AvaTax including where the company code is entered, where the credentials are entered and where tax codes can be mapped within the application.

#### **Address Validation Implemented**

An Address Validation utility is present in the application.

## **2. CUSTOMER RECORD INTEGRATION:**

#### **Customer Code**

Identify customer code (number, ID) to pass to the AvaTax service.

#### **Entity Use Code**

This is a group of codes that indicate the type of exemption. See the standard codes, but be aware that users are able to create custom codes as well. It is best to manage this value in your application's Customer record and pass it to AvaTax as CustomerUsageType at either the document or line level, whichever is applicable.

## **3. ITEMS/CHARGE INTEGRATION:**

#### **Item Code**

Identify customer code (number, ID) to pass to the AvaTax service.

#### **Item Description**

Identify item/service/charge description to pass to the AvaTax service with a human-readable description or item name.

#### **AvaTax Tax Code mapping - Item Code/SKU**

Association of an item or item group to an AvaTax Tax Code to describe the taxability (e.g. Clothing-Shirts – B-to-C).

#### **Non-Standard Items**

Association of Freight/Shipping and/or Handling charge to an AvaTax Tax Code to describe the taxability.

## **4. SALES/BILLING DOCUMENTATION INTEGRATION:**

#### **DocCode**

Values that can come across to AvaTax as the DocCode.

#### **CustomerCode**

Values that can come across to AvaTax as the Customer Code.

#### **DocDate**

Value that comes across to AvaTax as the DocDate.

#### **TaxCalculation Date**

Value that is used for Tax Calculation Date in AvaTax.

#### **DocType**

DocType used for varying stages of the transaction life cycle.

#### **Header Level - Destination Address**

Value that is sent to AvaTax for Destination Address at the header level.

#### **Header Level - Origin Address**

Value that is sent to AvaTax for Origin Address at the header level.

#### **Line Number**

Value that is sent to AvaTax for the Line Number.

#### **Item Code**

Value that is sent to AvaTax for the Item Code.

#### **Item Description**

Value that is sent to AvaTax for the Item Description.

#### **Quantity**

Value that is sent to AvaTax for the Quantity.

**Amount (extended)**

Value that is sent to AvaTax for the Amount.

**Tax Code - Required**

Value that is sent to AvaTax for the Tax Code.

**Verify Avalara Calculated Tax Amount is used in transaction**

Confirm that the integration is taking the Tax Amount value from the GetTax response and using that to decorate the tax fields in the platform. **Do NOT take our rate and use it to calculate the tax within the platform itself.**

**Freight items must be transmitted separately**

Freight Items must be sent to AvaTax as a separate line item with a configurable tax code.

**GetTax call – Sales Order/Sales Invoices**

Ensure that invoices are processed through a logical document lifecycle.

**PostTax/CommitTax Call – Invoices**

Ensure that invoices are committed/posted for reporting appropriately.

**Send discounts appropriately**

Standard discounts included in line-level extended amount, manufacturer's coupons and hostess credits transmitted as additional line items.

## 5. SERVER AUDIT CLARITY AND INSTALLATION REQUIREMENTS:

**Pass connector identifier information via the TaxSvc.Profile.Client property**

Integrations must include information about the connector, such as name, version, and company name, as a signature to each transaction.

EXAMPLE: TaxSvc.Profile.Client = "Dynamics AX,9.0,MyApp for AX by ACME INC,1.0"

**Reasonable errors on server-side analysis**

There should be no errors except those that would result from normal (but invalid) user input (e.g. invalid address data). Such errors must be logged/displayed appropriately to the application.

**Reasonable ratio of GetTax and address validation calls to committed documents**

In a normal workflow, we expect to see (on average, including abandoned carts) up to 10 tax calculations per finalized document. In a straight-forward order entry process, the number of calls should be about three to five.

**Demonstrate and document installation of software – Install Shield or equivalent where applicable**

Customers should have an easy and trouble-free installation of the software.