



Avalara AvaTax for

AX 2012 R3 CU09 Retail

Code Change Document - EPOS customisations

Version 1.0

Revision date: 27/02/18

Product release #: Avalara AvaTax for AX 2012 R3 CU09 Retail Version AX 6.3.2000.326.01.00



Avalara may have patents, patent applications, trademarks, copyrights, or other intellectual property rights governing the subject matter in this document. Except as expressly provided in any written license agreement from Avalara, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2013 Avalara, Inc. All rights reserved.

Avalara, AvaTax, AvaTax Calc, AvaTax Returns, AvaTax Certs, AvaTax Local, AvaLocal, AvaTax POS,

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

Prerequisite:

Destination folder should contain Microsoft.Dynamics.Retail.Pos.Contracts.dll with relevant version

Installation Steps:

Add reference of following two files in VS project:

- 1) **Avalara.Ax.Retail.dll**
- 2) **Avalara.AvaTax.Adapter.dll**
- 3) **Code changes in shared projects**

AvaTax Changed the following list of Projects in existing Retail SDK projects

Project/Form Name	Methods	Code Changes
Tax.cproj	Project → Properties → Setting	<p>Set value of ProviderAssemblies as</p> <p>*</p> <p>Extension\Avalara.Ax.Retail.dll</p> <p>Set value of ProviderClassNames as</p> <p>*</p> <p>Avalara.Ax.Retail.AvaTaxProvider</p>
Tax.cproj	ProviderLoader.cs/GetProviderDefinitions	<p>Search for the following</p> <p>StringCollection assemblies = Microsoft.Dynamics.Retail.Pos.Tax.Properties.Settings.Default.ProviderAssemblies;</p> <p>StringCollection classes = Microsoft.Dynamics.Retail.Pos.Tax.Properties.Settings.Default.ProviderClassNames;</p> <p>and replace following line of codes</p> <p>StringCollection assemblies =Microsoft.Dynamics.Retail.Pos.TaxService.Properties.Settings.Default.ProviderAssemblies;</p> <p>StringCollection classes =Microsoft.Dynamics.Retail.Pos.TaxService.Properties.Settings.Default.ProviderClassNames;</p>
BlankOperations.cproj		<p>Please add the following reference to the project</p> <p>Avalara.AvaTax.Adapter.dll</p> <p>Avalara.Ax.Retail.dll</p>

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

<p>BlankOperations.cpro</p>	<p>BlankOperations.cs</p>	<p>Search for the following</p> <pre>using Microsoft.Dynamics.Retail.Pos.Contracts.Services;</pre> <p>Add the following lines after that.</p> <pre>using System; using Avalara.Ax.Retail.Util; using Avalara.Ax.Retail.Diagnostics;</pre>
<p>BlankOperations.cpro</p>	<p>BlankOperations.cs/ BlankOperations</p>	<p>Search for the following lines of code</p> <pre>StringBuilder comment = new StringBuilder(128); comment.AppendFormat(ApplicationLocalizer.Language.Translate(50700) , operationInfo.OperationId); comment.AppendLine(); comment.AppendFormat(ApplicationLocalizer.Language.Translate(50701) , operationInfo.Parameter); comment.AppendLine(); comment.Append(ApplicationLocalizer.Language.Translate(50702)); using (LSRetailPosis.POSProcesses.frmMessage dialog = new LSRetailPosis.POSProcesses.frmMessage(comment.ToString(), MessageBoxButtons.OK, MessageBoxIcon.Error)) { LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(dialog) ; } And replace with the following lines of code</pre> <pre>if (operationInfo.OperationId == "9999") { //LSRetailPosis.POSProcesses.WinControls.StringPad switch (operationInfo.Parameter.ToUpper()) { case "ECMS": //Sets exemption number, scope of this number will be limited for current transaction. using (LSRetailPosis.POSProcesses.frmInputNumpad inputDialog = new LSRetailPosis.POSProcesses.frmInputNumpad()) { inputDialog.EntryTypes = Contracts.UI.NumpadEntryTypes.None; inputDialog.MaxNumberOfDigits = 25; inputDialog.PromptText =</pre>

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

```
ApplicationLocalizer.Language.Translate("Enter Exemption no:", -1);

        if
(!string.IsNullOrEmpty(TaxSvcHelper.exemptionNumber))
        {
            inputDialog.InputText =
TaxSvcHelper.exemptionNumber;
        }

LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(inputD
ialog);

        if (inputDialog.DialogResult == DialogResult.OK)
        {
            if (inputDialog.InputText.Length > 25) //Client side
validation for ECMS
            {

//InternalApplication.Services.Dialog.ShowMessage(
//    "ExemptionNo lenght must be between 0 and 25
characters.", MessageBoxButtons.OK,
//    MessageBoxIcon.Error);
            using (LSRetailPosis.POSProcesses.frmMessage
dialog = new LSRetailPosis.POSProcesses.frmMessage("ExemptionNo
length must be between 0 and 25 characters.", MessageBoxButtons.OK,
MessageBoxIcon.Error))
            {

LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(dialog)
;
            }
        }
        else
        {
            TaxSvcHelper.exemptionNumber =
inputDialog.InputText;
        }
    }
    break;
    case "EUCODE":

        //Sets entity use code, scope of this number will be limited
for current transaction.
        using (LSRetailPosis.POSProcesses.frmInputNumpad
inputDialog = new LSRetailPosis.POSProcesses.frmInputNumpad())
        {
            inputDialog.EntryTypes =
Contracts.UI.NumpadEntryTypes.None;
            inputDialog.MaxNumberOfDigits = 40;
            inputDialog.PromptText =
ApplicationLocalizer.Language.Translate("Enter Entity Use Code:", -2);

            if (!string.IsNullOrEmpty(TaxSvcHelper.entityUseCode))
            {
```

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre> inputDialog.InputText = TaxSvcHelper.entityUseCode; } LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(inputDialog); if (inputDialog.DialogResult == DialogResult.OK) { if (inputDialog.InputText.Length > 40) //Client side validation for Entity use code { //InternalApplication.Services.Dialog.ShowMessage(// "EntityUseCode length must be between 0 and 25 characters.", MessageBoxButtons.OK, // MessageBoxIcon.Error); using (LSRetailPosis.POSProcesses.frmMessage dialog = new LSRetailPosis.POSProcesses.frmMessage("EntityUseCode length must be between 0 and 25 characters.", MessageBoxButtons.OK, MessageBoxIcon.Error)) { LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(dialog) ; } } else { TaxSvcHelper.entityUseCode = inputDialog.InputText; } } break; case "DIAGNOSTICS": Diagnostics diagnostics = new Diagnostics(); diagnostics.ShowDialog(); break; default: break; } } else { StringBuilder comment = new StringBuilder(128); comment.AppendFormat(ApplicationLocalizer.Language.Translate(50700) , operationInfo.OperationId); comment.AppendLine(); comment.AppendFormat(ApplicationLocalizer.Language.Translate(50701) , operationInfo.Parameter); comment.AppendLine(); }</pre>
--	--	---

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre> comment.Append(ApplicationLocalizer.Language.Translate(50702)); using (LSRetailPosis.POSProcesses.frmMessage dialog = new LSRetailPosis.POSProcesses.frmMessage(comment.ToString(), MessageBoxButtons.OK, MessageBoxIcon.Error)) { LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(dialog) ; } } </pre>
Customer.cproj		<p>Please add the following reference to the project</p> <p>Avalara.AvaTax.Adapter.dll Avalara.Ax.Retail.dll</p>
Customer.cproj	frmNewCustomer.cs/ SaveCustomer	<p>Please search for the following line</p> <p>using DM = Microsoft.Dynamics.Retail.Pos.DataManager;</p> <p>Add the following lines after that.</p> <pre> using System.Data.SqlClient; using Avalara.AvaTax.Adapter.AddressService; using Avalara.Ax.Retail; using Avalara.Ax.Retail.Util; //using LSRetailPosis.Transaction; //using Microsoft.Dynamics.Retail.Pos.Contracts; using AvaAddress = Avalara.AvaTax.Adapter.AddressService.Address; using System.Collections.ObjectModel; </pre>
Customer.cproj	frmNewCustomer.cs/ SaveCustomer	<p>Please search for the following line</p> <p>Note : There could be multiple occurrence present in the function with the below string . Please consider the 1st occurrence.</p> <pre> IList<Int64> entityKeys = new List<Int64>(); </pre> <p>Add the following lines after that.</p> <pre> //AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableAddress Validation && TaxSvcHelper.countryListForValidation.Contains(Convert.ToString (this.addressViewModel.Address.Country))) { AddressSvcWrapper addressSvc = new AddressSvcWrapper(); ValidateRequest validateRequest = new ValidateRequest(); validateRequest.Address = new AvaAddress(); </pre>

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre> validateRequest.Address.Line1 = this.addressViewModel.Address.StreetName; validateRequest.Address.City = this.addressViewModel.Address.City; validateRequest.Address.Region = this.addressViewModel.Address.State; validateRequest.Address.PostalCode = this.addressViewModel.Address.PostalCode; validateRequest.Address.Country = this.addressViewModel.Address.Country; ValidationResult validateResult = addressSvc.Validate(validateRequest); DialogResult ValidateAddressPrompt; if (validateResult.ResultCode == Avalara.AvaTax.Adapter.SeverityLevel.Success) { if (validateResult.Addresses.Count > 0) { string validateAddressMsg = string.Format("Address Validated by AVATAX{0}{1} {2} {3} {4} {5} {6} {7}", Environment.NewLine, validateResult.Addresses[0].Line1, validateResult.Addresses[0].City, validateResult.Addresses[0].Region, validateResult.Addresses[0].PostalCode, validateResult.Addresses[0].Country, Environment.NewLine, "DO you want to update?"); ValidateAddressPrompt = Pos.Customer.Customer.InternalApplication.Services.Dialog.ShowM essage(validateAddressMsg, MessageBoxButtons. YesNo, MessageBoxIcon. Question); if (ValidateAddressPrompt == DialogResult.Yes) { this.addressViewModel.Address.StreetName = validateResult.Addresses[0].Line1; this.addressViewModel.Address.City = validateResult.Addresses[0].City; this.addressViewModel.Address.State =</pre>
--	--	--

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre>validateResult.Addresses[0].Region; this.addressViewModel.Address.PostalCode = validateResult.Addresses[0].PostalCode; //TODO: need to write mapping code between Avalara country code and AX country code. //this.addressViewModel.Address.Country = validateResult.Addresses[0].Country; } } else { string validateAddressErrMsg = string.Format("{0} {1} {2}", Environment.NewLine, validateResult.Messages[0].Name, validateResult.Messages[0].Details); Pos.Customer.Customer.InternalApplication.Services.Dialog.ShowM essage(validateAddressErrMsg, MessageBoxButtons.OK, MessageBoxIcon.Information); } System.Data.SqlClient.SqlConnection sqlDataConnection = Pos.Customer.Customer.InternalApplication.Settings.Database.Con nection; string sqlString = "select [RETAILTAXGROUPID] from [ax].[AVA_CONFIGURATION]"; System.Data.SqlClient.SqlCommand sqlCommand = new SqlCommand(sqlString, sqlDataConnection); sqlDataConnection.Open(); System.Data.SqlClient.SqlDataReader sqlReader = sqlCommand.ExecuteReader(); if (sqlReader.HasRows) { sqlReader.Read(); strAvaTaxGroup = sqlReader[0].ToString(); } } //AVA-</pre>
--	--	---

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

<p>Customer.cproj</p>	<p>frmNewCustomer.cs/ SaveCustomer</p>	<p>Please search for the following line Note : There could be multiple occurrence present in the function with the below string . Please consider the 1st occurrence.</p> <pre> if (createdAx) { </pre> <p>Add the following lines after that.</p> <pre> //AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableAddress Validation && !Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalcu lation) { ReadOnlyCollection<object> carr; carr = Pos.Customer.Customer.InternalApplication.TransactionServices.I nvokeExtension("ava_updateCustomer", tempCustomer.CustomerId); } //AVA- </pre>
<p>Customer.cproj</p>	<p>frmNewCustomer.cs/ SaveCustomer</p>	<p>Please search for the following line</p> <pre> createdLocal = customerDataManager.SaveTransactionalCustomer((tempCustomer as LSRetailPosis.Transaction.Customer), entityKeys); </pre> <p>Add the following lines after that.</p> <pre> //AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableAddress Validation && !Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalcu lation) { </pre> <pre> this.viewModel.Customer.SalesTaxGroup = strAvaTaxGroup; this.addressViewModel.Address.SalesTaxGroup = strAvaTaxGroup; } //AVA- </pre>
<p>Customer.cproj</p>	<p>ViewModel\AddressVi ewModel.cs</p>	<p>Please search for the following line</p> <pre> using DM = Microsoft.Dynamics.Retail.Pos.DataManager; </pre> <p>Add the following lines after that.</p> <pre> using System.Data.SqlClient; using AvaAddressService = Avalara.AvaTax.Adapter.AddressService; using Microsoft.Dynamics.Retail.Pos.Contracts; using AvaAddress = Avalara.AvaTax.Adapter.AddressService.Address; using System.Collections.ObjectModel; using Avalara.Ax.Retail.Util; </pre>

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<p>using Avalara.Ax.Retail;</p>
<p>Customer.cproj</p>	<p>ViewModel\AddressViewModel.cs\ SaveShippingAddress</p>	<p>Please search for the following line</p> <p>IAddress tempAddress = this.Address;</p> <p>Add the following lines after that.</p> <pre style="color: red;"> //AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableAddress Validation && TaxSvcHelper.countryListForValidation.Contains(Convert.ToString (this.Address.Country))) { try { AddressSvcWrapper addressSvc = new AddressSvcWrapper(); AvaAddressService.ValidateRequest validateRequest = new AvaAddressService.ValidateRequest(); validateRequest.Address = new Avalara.AvaTax.Adapter.AddressService.Address(); validateRequest.Address.Line1 = this.Address.StreetName; validateRequest.Address.City = this.Address.City; validateRequest.Address.Region = this.Address.State; validateRequest.Address.PostalCode = this.Address.PostalCode; validateRequest.Address.Country = this.Address.Country; AvaAddressService.ValidateResult validateResult = addressSvc.Validate(validateRequest); DialogResult ValidateAddressPrompt; if (validateResult.ResultCode == Avalara.AvaTax.Adapter.SeverityLevel.Success) { if (validateResult.Addresses.Count > 0) { string validateAddressMsg = string.Format("Address </pre>

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre>Validated by AVATAX{0}{1} {2} {3} {4} {5} {6} {7}", Environment.NewLine, validateResult.Addresses[0].Line1, validateResult.Addresses[0].City, validateResult.Addresses[0].Region, validateResult.Addresses[0].PostalCode, validateResult.Addresses[0].Country, Environment.NewLine, "DO you want to update?"); ValidateAddressPrompt = Pos.Customer.Customer.InternalApplication.Services.Dialog.ShowM essage(validateAddressMsg, MessageBoxButtons. YesNo, MessageBoxIcon. Question); if (ValidateAddressPrompt == DialogResult.Yes) { this.Address.StreetName = validateResult.Addresses[0].Line1; this.Address.City = validateResult.Addresses[0].City; this.Address.State = validateResult.Addresses[0].Region; this.Address.PostalCode = validateResult.Addresses[0].PostalCode; } } } else { string validateAddressErrMsg = string.Format("{0} {1} {2}", Environment.NewLine, validateResult.Messages[0].Name, validateResult.Messages[0].Details); Pos.Customer.Customer.InternalApplication.Services.Dialog.ShowM essage(validateAddressErrMsg, MessageBoxButtons.OK, MessageBoxIcon.Information); } if (string.IsNullOrEmpty(this.Address.SalesTaxGroup)) {</pre>
--	--	--

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre> System.Data.SqlClient.SqlConnection sqlDataConnection = Pos.Customer.Customer.InternalApplication.Settings.Database.Con nection; string sqlString = "select [RETAILTAXGROUPID] from [AX].[AVA_CONFIGURATION]"; System.Data.SqlClient.SqlCommand sqlCommand = new SqlCommand(sqlString, sqlDataConnection); sqlDataConnection.Open(); System.Data.SqlClient.SqlDataReader sqlReader = sqlCommand.ExecuteReader(); string strAvaTaxGroup = string.Empty; if (sqlReader.HasRows) { sqlReader.Read(); strAvaTaxGroup = sqlReader[0].ToString(); this.Address.SalesTaxGroup = strAvaTaxGroup; } } catch (Exception ex) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Inside SaveShippingAddress, error : " + ex.Message); } } //AVA-</pre>
TransactionTrigger s.cproj		<p>Please add the following reference to the project</p> <pre> Avalara.AvaTax.Adapter.dll Avalara.Ax.Retail.dll Microsoft.Dynamics.Retail.Pos.SystemCore.dll</pre>
TransactionTrigger s.cproj	TransactionTriggers.c s	<p>Please search for the following line</p> <pre> using Microsoft.Dynamics.Retail.Pos.Contracts.Triggers;</pre> <p>Add the following lines after that.</p> <pre> using System.ComponentModel.Composition; using System.Data.SqlClient; using LSRetailPosis; using Microsoft.Dynamics.Retail.Pos.Contracts.DataEntity; using Microsoft.Dynamics.Retail.Pos.Contracts.Triggers; using Avalara.Ax.Retail.Util; using System; using System.Collections.ObjectModel;</pre>

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre>using LSRetailPosis.Transaction; using Microsoft.Dynamics.Retail.Pos.SystemCore;</pre>
TransactionTrigger s.cproj	TransactionTriggers.c s/ BeginTransaction	<p>Please search for the following line</p> <pre>LSRetailPosis.ApplicationLog.Log("TransactionTriggers.BeginTransaction", "When starting the transaction...", LSRetailPosis.LogTraceLevel.Trace);</pre> <p>Add the following lines after that</p> <pre>//AVA+ TaxSvcHelper.isReturnTran = false; //AVA-</pre>
TransactionTrigger s.cproj	TransactionTriggers.c s/ PreEndTransaction	<p>Please search for the following line</p> <pre>LSRetailPosis.ApplicationLog.Log("TransactionTriggers.PreEndTransaction", "When concluding the transaction, prior to printing and saving...", LSRetailPosis.LogTraceLevel.Trace);</pre> <p>Add the following lines after that</p> <pre>//AVA++ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalculation) { if (posTransaction is RetailTransaction) { RetailTransaction retailTransaction = (RetailTransaction)posTransaction; TaxSvcHelper.SetInvoiceComment(retailTransaction); } } //AVA-</pre>
TransactionTrigger s.cproj	TransactionTriggers.c s/ PostEndTransaction	<p>Please search for the following line</p> <pre>LSRetailPosis.ApplicationLog.Log("TransactionTriggers.PostEndTransaction", "When concluding the transaction, after printing and saving", LSRetailPosis.LogTraceLevel.Trace);</pre> <p>Add the following lines after that</p> <pre>//AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalculation) { if (posTransaction is RetailTransaction) { if (TaxSvcHelper.exemptionNumber !=</pre>

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre>String.Empty) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Cleared Exemption Number"); TaxSvcHelper.exemptionNumber = string.Empty; } if (TaxSvcHelper.entityUseCode != string.Empty) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Cleared EntityUseCode"); TaxSvcHelper.entityUseCode = string.Empty; } if (TaxSvcHelper.retailTransactionTrace != null) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Reset TaxOverride Date"); } } if (posTransaction is RetailTransaction) { if (TaxSvcHelper.exemptionNumber != string.Empty) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Cleared Exemption Number"); //Exemption number scope ends here. TaxSvcHelper.exemptionNumber = string.Empty; } if (TaxSvcHelper.entityUseCode != string.Empty) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Cleared EntityUseCode"); TaxSvcHelper.entityUseCode = string.Empty; } if (TaxSvcHelper.retailTransactionTrace != null) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Reset TaxOverride Date"); } } if (posTransaction is CustomerOrderTransaction)</pre>
--	--	---

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre> { //Question - Do we need to commit this transaction in AvaTax? } } } //AVA- </pre>
TransactionTrigger s.cproj	TransactionTriggers.c s/ PostVoidTransaction	<pre> //AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalc ulation) { string source = "TransactionTriggers.PostVoidTransaction"; string value = "After voiding the transaction..."; LSRetailPosis.ApplicationLog.Log(source, value, LSRetailPosis.LogTraceLevel.Trace); LSRetailPosis.ApplicationLog.WriteAuditEntry(source, value); } //AVA- </pre>
TransactionTrigger s.cproj	TransactionTriggers.c s/ PreReturnTransaction	<p>Please search for the following line</p> <pre> LSRetailPosis.ApplicationLog.Log("TransactionTriggers.PreReturnTransac tion", "Before returning the transaction...", LSRetailPosis.LogTraceLevel.Trace); </pre> <p>Add the following lines after that</p> <pre> //AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalc ulation) { try { TaxSvcHelper.isReturnTran = true; if (originalTransaction != null) { if (!string.IsNullOrEmpty(Convert.ToString(((LSRetailPosis.Transac tion.PosTransaction)(originalTransaction)).ReceiptId))) { SetAvaAddressForReturnCal(Convert.ToString(((LSRetailPosis.Tran saction.PosTransaction)(originalTransaction)).ReceiptId) , Convert.ToString(((LSRetailPosis.Transaction.PosTransaction)(or iginalTransaction)).StoreId) , Convert.ToString(((LSRetailPosis.Transaction.PosTransaction)(or iginalTransaction)).TerminalId)); } } } } //AVA- </pre>

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre> ReadOnlyCollection<object> retHQObjArr; retHQObjArr = PosApplication.Instance.TransactionServices.InvokeExtension("ava_findallAddresses", ((LSRetailPosis.Transaction.RetailTransaction)(originalTransaction)).ReceiptId, ((LSRetailPosis.Transaction.RetailTransaction)(originalTransaction)).StoreId, ((LSRetailPosis.Transaction.RetailTransaction)(originalTransaction)).TerminalId, ((LSRetailPosis.Transaction.RetailTransaction)(originalTransaction)).TransactionId); if (retHQObjArr.Count > 0) { if (Convert.ToBoolean(retHQObjArr[1])) { TaxSvcHelper.OriginalAllLineShippingAddress = Convert.ToString(retHQObjArr[2]); } } } catch (Exception ex) { TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("In PreReturnTransaction Error : ", ex.Message)); } } //AVA- </pre>
<p>TransactionTriggers.s.cproj</p>	<p>TransactionTriggers.cs/</p>	<p>Please go to end of file and search for the 2nd last occurrence of "}" char. And following lines of code . Note : It is nothing but just adding 2 new function in the file</p> <pre> Private void SetAvaAddressForReturnCal(string receiptId, string storeId, string terminalId) { try { TaxSvcHelper.OriginalstoreAddress = null; TaxSvcHelper.OriginalShippingAddress = null; string strconnect = LSRetailPosis.Settings.ApplicationSettings.Database.LocalConnection.ConnectionString; System.Data.SqlClient.SqlConnection sqlDataConnection = new SqlConnection(); </pre>

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre> sqlDataConnection.ConnectionString = strconnect; try { sqlDataConnection.Open(); string sqlString = "select [COMMENT], [transDate] from [dbo].[RETAILTRANSACTIONTABLE] where receiptid= '" + receiptId + "'"; System.Data.SqlClient.SqlCommand sqlCommand = new SqlCommand(sqlString, sqlDataConnection); System.Data.SqlClient.SqlDataReader sqlReader = sqlCommand.ExecuteReader(); string strComment = string.Empty; if (sqlReader.HasRows) { sqlReader.Read(); strComment = sqlReader[0].ToString(); TaxSvcHelper.OriginalDocDate = Convert.ToDateTime(sqlReader[1]); if (!string.IsNullOrEmpty(strComment)) { Avalara.AvaTax.Adapter.AddressService.Address tmpAddress = TaxSvcHelper.SpiltAddressstring(strComment); if(tmpAddress != null) { if (!(string.IsNullOrEmpty(tmpAddress.Line1)) !(string.IsNullOrEmpty(tmpAddress.PostalCode)) !(string.IsNullOrEmpty(tmpAddress.City)) !(string.IsNullOrEmpty(tmpAddress.Region)) !(string.IsNullOrEmpty(tmpAddress.Country))) { if (TaxSvcHelper.OriginalShippingAddress == null) TaxSvcHelper.OriginalShippingAddress = TaxSvcHelper.CreateAddress(); TaxSvcHelper.OriginalShippingAddress = tmpAddress; } } //Added for return case TaxSvcHelper.exemptionNumber = TaxSvcHelper.SpiltECMSstring(strComment); TaxSvcHelper.entityUseCode = TaxSvcHelper.SpiltEUCodestring(strComment); TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("I n SetAvaAddressForReturnCal with exemptionNumber = {0}, entityUseCode = {1}", TaxSvcHelper.exemptionNumber, TaxSvcHelper.entityUseCode)); } } } }</pre>
--	--	--

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

		<pre> catch (Exception ex) { TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("I n SetAvaAddressForReturnCal-1 Error : ", ex.Message)); } if (TaxSvcHelper.OriginalShippingAddress == null) { AvaSetAddress(receiptId, storeId, terminalId); } if (string.IsNullOrEmpty(TaxSvcHelper.exemptionNumber) string.IsNullOrEmpty(TaxSvcHelper.entityUseCode)) { AvaSetECMSandEUCodeAddress(receiptId, storeId, terminalId); } } catch (Exception ex) { TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("I n SetAvaAddressForReturnCal-2 Error : ", ex.Message)); } } public void AvaSetECMSandEUCodeAddress(string receiptId, string storeId, string terminalId) { try { ReadOnlyCollection<object> retHQObjArr; retHQObjArr = Pos.SystemCore.PosApplication.Instance.TransactionServices.Invo keExtension("ava_getComment", receiptId, storeId, terminalId); if (retHQObjArr.Count > 0) { if (Convert.ToBoolean(retHQObjArr[1])) { TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("a va_getComment = ", retHQObjArr[2])); string strComment = Convert.ToString(retHQObjArr[3]); TaxSvcHelper.exemptionNumber = TaxSvcHelper.SpiltECMSstring(strComment); TaxSvcHelper.entityUseCode = TaxSvcHelper.SpiltEUCodestring(strComment); TaxSvcHelper.OriginalDocDate = Convert.ToDateTime(retHQObjArr[4]); TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("I n AvaSetECMSandEUCodeAddress with exemptionNumber = {0}, entityUseCode = {1}", TaxSvcHelper.exemptionNumber, TaxSvcHelper.entityUseCode));</pre>
--	--	--

Code Change Document

AvaTax Connector for AX POS 2012 R3 Retail CU09 EPOS (AX 6.3.2000.326.01.00)

```
    }
    }
    }
    catch (Exception ex)
    {
        TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("E
xception in AvaSetECMSandEUCodeAddress", ex.Message +
Environment.NewLine + ex.StackTrace));
    }

    }

    public void AvaSetAddress(string receiptId, string
storeId, string terminalId)
    {
        try
        {
            TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("I
n AvaSetAddress calling ava_getStoreAddress with receiptId =
{0}, storeId = {1}, terminalId = {2}", receiptId, storeId,
terminalId));
            ReadOnlyCollection<object> rethQObjArr;

            rethQObjArr =
Pos.SystemCore.PosApplication.Instance.TransactionServices.Invo
keExtension("ava_getStoreAddress", receiptId, storeId,
terminalId);
            //
            Pos.BlankOperations.BlankOperations.InternalApplication.Transac
tionServices.InvokeExtension("getStoreAddress", receiptId);
            if (rethQObjArr.Count > 0)
            {
                if (Convert.ToBoolean(rethQObjArr[1]))
                {
                    if (TaxSvcHelper.OriginalstoreAddress ==
null) TaxSvcHelper.OriginalstoreAddress =
TaxSvcHelper.CreateAddress();
                    TaxSvcHelper.OriginalstoreAddress =
TaxSvcHelper.SpiltAddressstring(Convert.ToString(rethQObjArr[3]
));
                }
            }
            TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("a
va_getStoreAddress return address = ", rethQObjArr[2]));
        }
    }
    catch (Exception ex)
    {
        TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("E
xception in AvaSetAddress", ex.Message + Environment.NewLine +
ex.StackTrace));
    }
}
```