



## Release Notes - Active Commerce 3.3

### What's New?

- Customer Address Books
- Persistent Shopping Carts
- Avalara AvaTax Plugin
- Multiple Payment Support
- Configurable Product Sorting
- Google Universal Analytics
- Moneris Payment Gateway
- Support for Coveo for Sitecore 4.0
- Various fixes and improvements, see details below

### Notes on Upgrading from Active Commerce 3.2

- If you are upgrading from an earlier version of Active Commerce, reference the Release Notes for Active Commerce 3.2 and 3.2 Update-1.
- After installing, you must update the database schema. To do this:
  - Go to the following address: <http://<site host>/sitecore/admin/databaseutility.aspx>
  - Click the “Update Schema” button
  - You should see a “Success!” message when completed
- Implementations which update order data via the *IOrderManager* or *IOrderRepository* will require minor updates. The Order Manager API has changed to better take advantage of NHibernate sessions and persistence ignorance. After querying and manipulating order data, you now simply need to call *Save()* on the *OrderManager* to flush any changes to the database. The *IOrderRepository* interface and implementation have a corresponding change, with a *Flush()* method added to the interface.
- Implementations which use a custom payment provider will require updates due to improvements made to the payments API. *CreditCardInfo* (via *PaymentDetails*) and payment *Amount* are now accessed on the *PaymentArgs*. Also, the use of *ITransactionData* is obsolete; transaction information should be set directly on the *TransactionDetails* property instead. See the [InvoicePayment](#) and [OnsitePayment](#) examples on our GitHub for updated examples of custom payment providers. *Note: if you are using one of the provided plugins (Authorize.NET, CyberSource, Moneris, or*

*PayPal*), an install of the latest plugin is all that is required.

- Implementations which extend *Sitecore.Ecommerce.Users.CustomerInfo* should update their classes to instead extend *ActiveCommerce.Users.CustomerInfo*. Active Commerce now includes its own implementation of this class which will now trigger change events in the cart, allowing promotions which trigger on changes to customer profile data.
- Existing implementations with saved customer data will need to migrate existing saved customer addresses into the new address datastore in the Active Commerce database. If you have extended the address data collected by Active Commerce, you may need to extend the new data model as well. The [migration project on our GitHub](#) contains example code for migrating address data. Be sure to extend this code as well if you have added new fields.
- Implementations which process transactions or complete checkout through custom code will require updates due to changes to how payments are added to the shopping cart. See the [Checkout via API example on our GitHub](#) for an updated example of setting the *PrimaryPayment* on the *ShoppingCart*.
- You'll need to modify your existing checkout page, adding the component which enables use of multiple addresses.
  - In Sitecore, open the checkout page (e.g. */sitecore/content/<site>/Home/shop/checkout*) in the Experience Editor
  - Select the *Checkout - Shipping Address* component, and click the *Create or edit personalization for this component* button
  - Check *Enable personalization of component design*
  - Add a new condition, named "Address Book"
    - Set the *Condition* to *where the current customer has saved addresses*
      - Note: If you do not see the "Customer" group in the Rule Set Editor, you'll need to perform the following:
        1. In Sitecore, go to */sitecore/system/Settings/Rules/Conditional Renderings/Tags/Default*
        2. Add "Customer" to the list of selected Tags
    - Set the *Personalize Component* to *Checkout - Shipping Address Book*
    - Set the *Personalize Content* to *Shipping Address Book* (create new content)

### Personalize the Component

Manage the personalization conditions, content, and design of the component. The list of conditions is prioritized. The first true condition determines which personalization content is displayed.

Enable personalization of component design. New Condition


Address Book
Actions ▾

**Condition**

where the current customer has saved addresses Edit


Hide Component


**Personalize Component:**



Checkout - Shipping x ...

**Personalize Content:**

 .../Shipping Address x ...




Shipping

Default
▾

If none of the other conditions are true, the default condition is used.

Hide Component

**Personalize Component:**



Checkout - Shipping Address ...

**Personalize Content:**

[Not set] Checkout ...

OK
Cancel

- Click OK and save.
- Repeat for the *Checkout - Billing Address* component.
- You'll also need to add the new *Address Form* item. This form is used within a modal when adding or editing a saved address, either from My Account or during checkout.
  - In Sitecore, go to `/sitecore/templates/Branches/ActiveCommerce/Active Commerce Website Branch/$name/Home/shop/account/address-input-modal`
  - Copy this item to the same location within your shop (e.g. `/sitecore/content/<site>/Home/shop/address-input-modal`)
- Make sure all changes are pushed through workflow and published.
- As with any Active Commerce upgrade, you should utilize WinMerge or another diff tool to analyze any changes in the base skin against any CSS, Javascript, or Sublayouts which you have skinned.

## New Features and Improvements

### Customer Address Books

- [AC-1031] - Customers can now save multiple addresses and choose from them for shipping and billing during checkout. Address books can be managed from the My Account area as well as the checkout process. Customer addresses are now stored in the Active Commerce database.
- [AC-1079] - “Company” is now collected as a default address field, with a rendering parameter available to hide the field if desired.

### Persistent Shopping Carts

- [AC-575] - Shopping cart persistence is now available as a fully supported plugin. By default cart data is persisted to the Active Commerce database. Anonymous customer carts are restored based on a Customer ID stored in a persistent cookie, and logged in customer carts are restored based on the Customer ID on the customer profile.

### Configurable Product Sorting

- [AC-940] - The available sorting options for a product list can now be configured on each category page. Sorting options can be extended and are either index-driven or dynamic.

### Plugins

- [AC-938] - Avalara AvaTax is now available as an official plugin for tax calculation. Customers utilizing pre-releases of this plugin on 3.2 or 3.2 Update-1 are strongly encouraged to upgrade to 3.3.
- [AC-937] - The Moneris payment gateway is now available as a payment provider plugin.
- [AC-1077] - Updated version of PayPal SDK to support TLS 1.2.

### Checkout and Orders

- [AC-744] - Multiple payments on a single cart are now supported, allowing for custom implementations and integrations of gift cards, loyalty points, etc.
- [AC-1021] - Added an example gift card integration on our GitHub.
- [AC-935] - Checkout and order processing will now properly handle an order with a \$0 total.
- [AC-908] - Now storing and displaying more information about the shipping option with orders.
- [AC-922] - It's now possible to manipulate the state of checkout steps via Javascript.
- [AC-843] - Configuration settings are now available to add a prefix or suffix to generated order numbers.
- [AC-844] - Improved NHibernate session handling for orders to improve performance

and decrease SQL load during order processing.

- [AC-933] [AC-804] [AC-958] - Improved background color configuration for built-in order receipt.

## Product Catalog

- [AC-785] - Now easier to get all the products in the product repository
- [AC-902] - Improved error messaging when previewing a product without a product code
- [AC-901] - Improved handling of security-restricted products returned by search queries

## Other Improvements

- [AC-656] - Upgraded built-in Google Analytics support to Universal Analytics
- [AC-1122] - Better handling of 500 errors returned by Cart and Checkout AJAX services
- [AC-143] - When using Persistent Carts, cart is emptied upon user sign out
- [AC-812] - Enabled fallback to base skin for configured image/media sizes
- [AC-825] - *RegistrationFinder* assembly scanning is now configurable to enable load time optimization

## Issues Resolved

- [AC-850] - Cart change event was not fired for customer email updates
- [AC-891] - AvaTax plugin pre-release always used *CustomerInfo* from current user
- [AC-896] - Shipping providers couldn't be workflowed due to use of shared fields
- [AC-898] - Shipping provider was cleared from cart even when using *ICartEventsDisabler*
- [AC-899] - *Account-Logout.ascx* used wrong navigation link name
- [AC-910] - Virtual products didn't render properly in the Shopping Cart
- [AC-921] - Checkout steps other than Shipping Address didn't activate when they were the first step
- [AC-928] - *GetDatasourceDependencies* pipeline processor needed an additional null check
- [AC-932] - Product delete was not removing from product from index in Sitecore 7.5
- [AC-948] - Shipping options would disappear entirely when one option was configured with an unresolvable service code
- [AC-970] - Sherpa's *Return Customer* campaign had a Display Name of "\$name" on Sitecore 8.1
- [AC-1041] - Default country list included The Netherlands twice
- [AC-1093] - Product package didn't install properly on some versions of Sitecore due to item name validation
- [AC-1094] - Creating a new AC site would fail when using SQL sessions in CM environment
- [AC-1150] - *Totals* objects were not serializable and could not be stored in session