

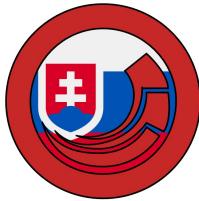
# SXA Storefront

Peter Prochazka

# Sitecore SXA Storefront Documentation

Compiled from [official Sitecore SXA Storefront documentation](#).

About author:



Peter Prochazka ([@chorpo](#) / [tothecore.sk](#))

Version 1.0 / 5th July 2018

More Sitecore guidelines and Sitecore related topics can be found on my blog [tothecore.sk](#).

You can find them also directly [in my github repositories](#).

# Table of Contents

## [The SXA Storefront Template](#)

[Introducing the Sitecore Experience Accelerator Storefront template](#)

[Overview of functionality in the SXA Storefront](#)

[Engagement functionality in the SXA Storefront](#)

[SXA Storefront themes](#)

[The SXA Storefront page templates, page designs, and partial designs](#)

[Security domains in SXA Storefront](#)

[SXA Storefront security roles](#)

## [Build a storefront with SXA Storefront](#)

[Set up the SXA Storefront template site](#)

[Create a new tenant and site with commerce features](#)

[Configure your storefront](#)

[The Commerce Control Panel settings for SXA Storefront](#)

[Configure Catalogs in SXA Storefront](#)

[Build a storefront with Commerce renderings for SXA](#)

[Add a language version to an existing storefront site](#)

[Configure the Buy Gift Card page](#)

## [Commerce renderings for SXA](#)

[Commerce Account renderings](#)

[Commerce Cart renderings](#)

[Commerce Catalog renderings](#)

[Commerce Checkout renderings](#)

[Commerce Orders renderings](#)

[Commerce Shared renderings](#)

[Commerce meta renderings](#)

## [Using Commerce renderings for SXA](#)

[Using rendering variants to display product information](#)

[Configure a product list and display it on your storefront](#)

## [Technical Overview](#)

[The SXA Storefront installation packages](#)

[Enforcing HTTPS on Commerce-related pages](#)

[Shop pages and canonical URLs](#)

[Caching SXA Storefront renderings](#)

## SXA Storefront architecture

## The SXA Storefront Template

### Introducing the Sitecore Experience Accelerator Storefront template

The Sitecore Experience Accelerator (SXA) Storefront template is an out-of-the-box storefront solution for building B2C e-commerce solutions. It has its own separate [installation packages](#) that contain a webshop template built with Commerce renderings for SXA, the Habitat catalog, and sample themes. The SXA Storefront enables you to quickly deploy storefronts on premise or in the Cloud.

There are several reasons to use the SXA Storefront, including:

- It is a full B2C retail storefront template based on Sitecore best practices, built on state-of-the-art technologies.
- The storefront can be adopted, skinned using SXA themes, and customized for production, or used as a best practice reference implementation.
- It takes advantage of [engagement functionality](#) in the Sitecore Experience Platform to track, act and follow up on visitor behavior to help create personal customer experiences and win customers for life.

## One connected experience

The SXA Storefront takes full advantage of the Sitecore Commerce Connect framework by:

- Collecting intelligence on customer behavior, registering outcomes, goals, and page events.
- Providing rules for personalizing the individual customer experience based on intelligence collected about customer behavior.

## Advantages over competitor offerings

The SXA Storefront offers advantages over competitors, including:

- The integration between the storefront and Sitecore Commerce uses Sitecore Commerce Connect, which is a standardized framework, and commerce API where engagement comes out-of-the-box.
- The storefront is based on state-of-the-art technologies including the latest Sitecore offerings in the form of Sitecore Experience Platform and

Sitecore Commerce.

### 3rd party Integration

To allow online payment during checkout, the Storefront includes integration with BrainTree. Contact BrainTree to negotiate an agreement and obtain the necessary merchant ID.

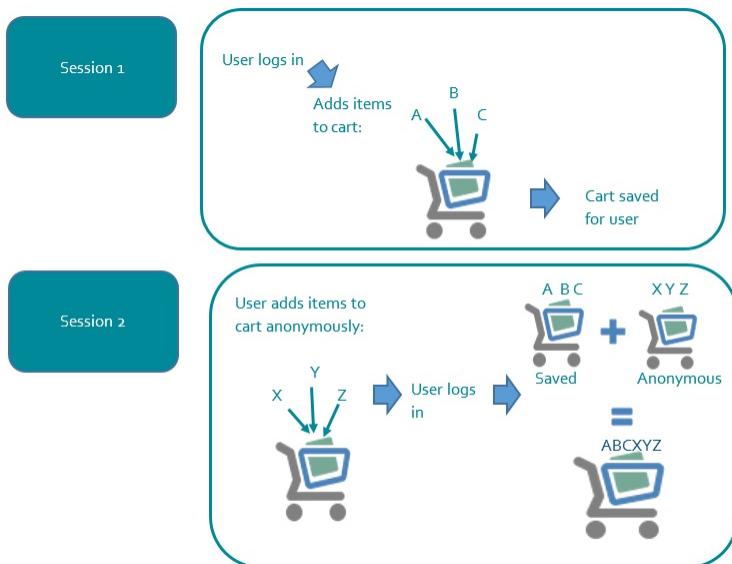
## Overview of functionality in the SXA Storefront

This topic provides you with an overview of functionality that is built into the SXA Storefront. The topic highlights functionality in the following areas:

- Shopping cart
- Catalog
- Product prices
- Product inventory
- Payment
- Shipping
- Multistore support
- Globalization
- Security
- Experience Editor support
- Mobile support
- Search engine support

### Shopping cart

Customers can add items they want to buy to a shopping cart on the storefront. If they leave the storefront before making a purchase, the items remain in the cart. An existing shopping cart is automatically resumed when a contact returns to the storefront.



*Session 1:* Customer logs in and adds items to cart, then logs out. Cart is saved for a later session.

*Session 2:* Customer browses the site and adds items to the cart anonymously. When the customer logs in, the items saved from Session 1 appear in the contact's shopping cart together with the items added anonymously in session 2.

## Catalog

Customers can browse the storefront by category and subcategory, and they can filter the results using facets.

Categories and products can have multiple different presentations within the same storefront, as well as across different storefronts that share the same catalog.

## Product prices

The storefront shows list prices for all products, and adjusted prices and savings in percentage for products on sale.

The storefront supports both indexed prices and dynamic price lookup in the commerce engine. List prices and adjusted prices exposed on product items via the data provider are automatically indexed. For search results, category pages, and product detail pages, the default is dynamic lookup.

## Product inventory

Search result and category pages show inventory statuses such as in-stock, out-of-stock, preorderable, and back-orderable.

Product detail pages show:

- Inventory status such as in-stock, out-of-stock, preorderable, and back-orderable.
- Number of items in stock.
- Date for when preorderable products will become available/in-stock.
- Date for when back-orderable products will become available/in-stock.

The storefront supports both cached inventory status information and dynamic inventory status lookup in the commerce engine. The default is dynamic lookup for both search results, category pages, and product detail pages.

## Payment

The following billing options are supported:

- Credit card
- Promotion codes
- Gift cards (can be redeemed as payment during checkout)

You configure payment options in the Commerce Control Panel for the individual storefronts.

## Shipping

The following shipping options are supported:

- Ship to address
- Send by email for digital products, including gift cards

Shipping options can be selected per shopping cart line.

You configure shipping methods in the Commerce Control Panel for the individual storefronts.

## Multistore support

Using SXA Storefront, you can host multiple storefronts within the same

Sitecore solution.

Individual stores can:

- Share the same catalog.
- Use different parts of the same catalog with exclusive as well as shared products.
- Use different catalogs.

## Globalization

Other than the product catalog, all textual content in Sitecore that is used on a given storefront is stored in connection with the individual storefront. When you use SXA, textual data that is stored on associated data items is stored in the *Data* folder unless it is meant to be shared across sites within a tenant.

This approach:

- Allows maximum flexibility in multistorefront scenarios where content might differ between shops.
- Keeps all content together in one area for easy content management.

System error messages are localizable and stored in the shared Commerce Control Panel for all storefronts.

## Security

By default, the storefront enforces use of the secure SSL based HTTPS protocol on pages and on asynchronous API calls that access sensitive data and content.

The storefront also enforces restrictions on browser caching and storage of data.

Unauthorized visitors cannot access APIs that handle sensitive data.

## Experience Editor support

In the Experience Editor, you can:

- Edit catalog data directly on the page, for example, category titles, and product information except prices, inventory status and stock count.
- Edit the association of images to products in the related product item.

- Edit most textual content on the page. Exceptions are hint text (placeholder or watermark text) and tooltips, which can be edited on the associated data item accessed from the floating toolbar.
- Execute a search and see the search result.

The Experience Editor looks similar to the final website, but some functionality is only available on the live storefront. For example, in the Experience Editor:

- Most forms cannot be submitted, for example, Login and Registration. When you work in the Experience Editor, you are logged in as a Sitecore user and it does not make sense to authenticate or register. This also applies to editing address and profile information.
- It is not possible to add products to the shopping cart.
- It is not possible to place an order or navigate the checkout using the regular buttons. The items that represent the pages in the checkout process must be manually selected from the Experience Editor breadcrumb.
- Renderings are always visible so you can select and edit them. An example is the Message Summary, which is not visible on the live storefront when it is empty.
- Catalog renderings show actual catalog data when the context is valid and they show placeholder data when the context is not valid.

## Mobile support

The SXA Storefront is built with responsive design for ease of use with mobile phones or tablets.

## Search engine support

Each category and each product has a single canonical URL for SEO purposes.

Even though the same categories and products can be presented in multiple different areas of the storefront, the metadata always points to the same canonical URL.

## Engagement functionality in the SXA Storefront

The SXA Storefront uses Sitecore Commerce Connect to integrate with the Sitecore Experience Platform to take full advantage of the engagement functionality built into the Sitecore Experience Platform.

When you use the SXA Storefront template, functionality for analytics and personalization is automatically set up and enabled.

### Analytics

The SXA Storefront inherits the [analytics functionality](#) that is built into Commerce Connect, and Sitecore registers:

- A *monetary outcome* as well as a goal when an order is submitted, where the monetary outcome is a dynamic value dependent on the order value.
- A *page event* when a category or product page is visited.
- A *page event* whenever one of the following entities is updated: Shopping Cart, Customer and User account, Inventory, and Gift Cart.
- A *search page event* whenever a search is executed.

### Personalization

The following personalization rule conditions are available:

- Shopping cart total amount compares to [Amount].
- Shopping cart total quantity compares to [QTY].
- Shopping cart contains product [P].
- The contact has triggered page event E within the last D days.
- Specific product P stock count compares to [QTY] (and by location).
- Current product stock count compares to [QTY] (and by location).
- Current product stock status equals [Status] (and by location).

## SXA Storefront themes

The SXA Storefront comes with a number of [themes](#) that support you through the design process. You can use them for inspiration when you [create new themes](#) for your site.

### Note

You must not modify any of the themes provided with the SXA Storefront solution. Any changes to the themes may be overwritten when the solution is updated.

This topic describes the intended use of each of the themes that are included with SXA and the SXA Storefront. With the different themes, Sitecore supports a design process where you start out planning the information architecture, functionality and navigation of the site, then create a mock-up with content, and finally create a storefront with a branded theme.

SXA and the SXA Storefront together come with the following themes:

- The Wireframe theme (SXA)
- The Storefront Mock-up themes (SXA Storefront)
- The Storefront Branded theme (SXA Storefront)
- Custom themes created with new commerce sites (SXA Storefront)

### The Wireframe theme (SXA)

The Wireframe theme is typically the theme that you would use when starting to design a new site. You are likely to use this theme while working on the information architecture, business objectives, user flows and primary usability issues. During this stage, you might share the site with colleagues for internal reviews and iterations.

### Note

The SXA Storefront does not support the Wireframe theme that comes with SXA. Therefore, when you create a new commerce storefront site, the *Storefront Mock-up with product images* theme is set as the default theme instead of *Wireframe*, which is normally assigned to new SXA sites.

However, you can create a [custom theme](#) with your SXA Storefront site to use for wireframing.

## The Storefront Mock-up themes (SXA Storefront)

The *Storefront Mock-up* themes are examples of the type of theme that you would use when you start updating your site with content such as texts, labels, and images.

SXA makes it easy for you to separate content from presentation, so you and your colleagues can work on content and design in parallel. This means that you can start adding content to the site as soon as the information architecture and the user flows have been determined.

The *Storefront Mock-up* themes are grayscale themes that give a more finished look to the site, while they are clearly not the final design.

You would typically create a mock-up theme in an iterative process as the overall design of the site begins to take shape. A designer in your company or an external design agency would use the mock-up theme as a starting point to create the final branded theme.

SXA Storefront comes with two mock up themes. The *Storefront Mock-up with product images* theme displays the original images from the catalog while the *Storefront Mock-up with mock images* theme displays generic gray scale images in place of the product images.

For example, the Registration rendering with either *Storefront Mock-up* theme looks like this:



A screenshot of a registration form. It contains three input fields: 'Email address\*' with placeholder 'Email address', 'Password\*' with placeholder 'Password', and 'Repeat password\*' with placeholder 'Password'. Below these fields is a button labeled 'Continue as guest'.

## The Storefront Branded theme (SXA Storefront)

The *Storefront Branded* theme is an example of a final design created by a design agency. It has a branded look and feel, and can serve as inspiration when you create a new branded theme for your company storefront.

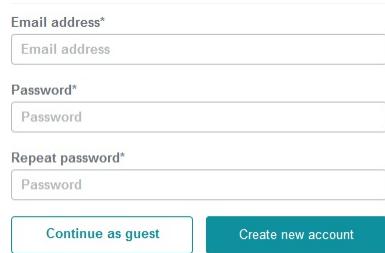
For example, the Registration rendering with the *Storefront Branded* theme looks like this in the Experience Editor:



The registration form for the *Storefront Branded* theme. It features three input fields: 'Email address\*', 'Password\*', and 'Repeat password\*'. Below the fields are two buttons: 'Continue as guest' (light blue) and 'Create new account' (dark teal).

Email address*	<input type="text"/>
Password*	<input type="password"/>
Repeat password*	<input type="password"/>
<a href="#">Continue as guest</a>	<a href="#">Create new account</a>

The Registration rendering with the *Storefront Branded* theme looks like this on the live SXA Storefront template site:

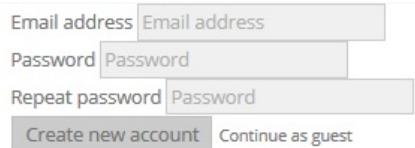


The registration form for a custom theme. It has the same three input fields: 'Email address\*', 'Password\*', and 'Repeat password\*'. It includes the same two buttons: 'Continue as guest' (light blue) and 'Create new account' (dark teal).

Email address*	<input type="text"/>
Password*	<input type="password"/>
Repeat password*	<input type="password"/>
<a href="#">Continue as guest</a>	<a href="#">Create new account</a>

Custom themes created with new commerce sites (SXA Storefront)  
When you [create a new commerce site](#), you can choose to create a new theme. The new theme includes support for the commerce renderings. The support does not include layout and the component styling is minimal. The custom theme can be used as a wireframe theme, or you can modify it.

For example, the Registration rendering with an un-styled custom theme looks like this:



The registration form for an un-styled custom theme. It has the same three input fields: 'Email address\*', 'Password\*', and 'Repeat password\*'. The buttons are also present but have a different visual style compared to the branded theme.

Email address	<input type="text"/>
Password	<input type="password"/>
Repeat password	<input type="password"/>
<a href="#">Create new account</a>	<a href="#">Continue as guest</a>

The theme includes all the component [Sass files](#). (Some of the files are empty.) This makes it easier to export the theme using [Creative Exchange](#).

## The SXA Storefront page templates, page designs, and partial designs

Most pages in the SXA Storefront are based on an SXA Storefront page template created specifically for the page. Each template has a [page design](#) associated with it, which is assigned by default to pages created with the template.

Each page design is made up of a number of partial designs.

In all folders within the *[My Storefront]/Home* folder, insert options are configured to let you efficiently add pages using the Storefront page designs.

This topic describes:

- Templates and page designs used for SXA Storefront pages
- Templates for additional storefront pages
- Partial designs used in SXA Storefront page designs

### Templates and page designs used for SXA Storefront pages

The following table lists the SXA Storefront pages, the template used to create each of them, and the page design associated with each page:

Storefront page	Template (in <i>/sitecore/templates/Project/&lt;tenant name&gt;</i> folder)	Page design
Home	Home	Default Commerce Page
Account Management	Account Management	Default Commerce Page
Address Book	Address Book	Default Commerce Page
Change Password	Change Password	Default Commerce

		Page
Edit Profile	Edit Profile	Default Commerce Page
My Order	My Order	Default Commerce Page
My Orders	My Orders	Default Commerce Page
Buy Gift Card	Buy Gift Card	Default Commerce Page
Category *	Category	Catalog Navigation Page
Checkout	Checkout	Checkout Page
Delivery	Delivery	Checkout Page
Billing	Billing	Checkout Page
Review	Review	Checkout Page
Order Confirmation	Order Confirmation	Checkout Page
About Storefront; Corporate Sales; Frequently Asked Questions; Jobs at Storefront	Content Page	Default Commerce Page
Forgot Password	Forgot Password	Default

		Commerce Page
Landing pages: DronePhoneWatch	Page	Default Commerce Page
Login	Login	Default Commerce Page
Product *	Product	Catalog Navigation Page
Registration	Registration	Default Commerce Page
Search	Sitecore/Page	Catalog Navigation Page
Shop *	Category	Catalog Navigation Page
Shopping Cart	Shopping Cart	Shopping Cart Page

#### Note

The SXA Storefront pages are designed to use the Bootstrap grid system. The Bootstrap grid system is applied automatically when you create the site. After you create the site, you must not select another grid system for any device in the Settings item for the site.

#### Templates for additional storefront pages

In addition to the templates used to create the pages in the SXA Storefront, the SXA Storefront comes with templates that you can use to create additional pages – one template for each of the following SXA Storefront page designs:

- Catalog Navigation Page
- Checkout Page
- Content Page
- Default Commerce Page
- Shopping Cart Page

These templates are in the `/sitecore/templates/Project/<tenant name>` folder.

### Insert Options

The SXA Storefront page templates are configured as insert options on items in the Home folder.

### Partial designs used in SXA Storefront page designs

The following table lists the partial designs used on each page design:

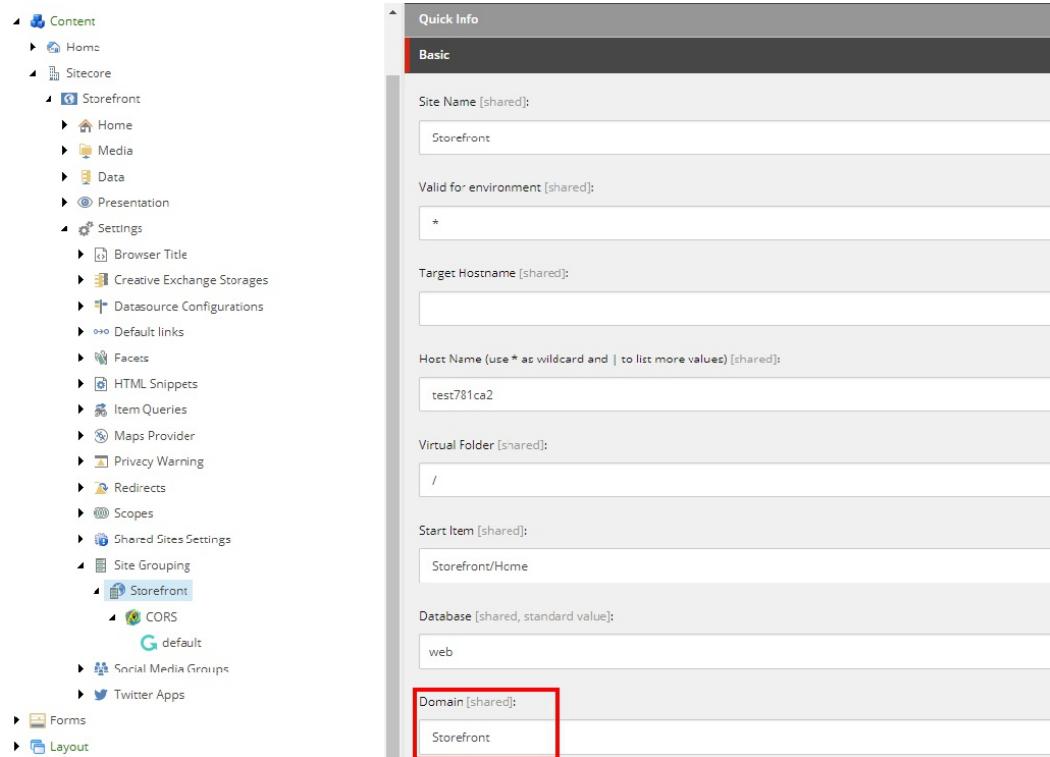
Page design	Partial designs
Catalog Navigation Page	Commerce Metadata for Catalog Items Main Content Default Commerce Header Default Commerce Footer Multi-Rows Base Header Structure Metadata
Checkout Page	Metadata Commerce Metadata Checkout Page Header Main Content Default Commerce Footer
Default Commerce Page	Main Content Default Commerce Footer Default Commerce Header Multi-Rows Base Header Structure Commerce Metadata Metadata

Shopping Cart Page

Main Content  
Default Commerce Footer  
Checkout Page Header  
Shopping Cart Page Header  
Commerce Metadata  
Metadata

## Security domains in SXA Storefront

When you create a site with SXA, you can configure the name of the security domain for the site in the Content Editor on the `/sitecore/content/[your tenant]/[your storefront]/Settings/Site Grouping/[your storefront]` item:



If you select the Commerce Foundation module when you create a tenant, and then you select the Commerce Storefront Components module when you create a site, then the Create a site wizard creates and configures a domain with the same name as the site, when it creates the site.

### Note

If you change the domain on a site that already has customer accounts, the customers will lose access to the site unless you update the domain on the accounts.

Sitecore Commerce Connect defines the `Sitecore.Commerce.Providers.iDomainProvider` interface. SXA Storefront includes an implementation of this provider interface in the SXA Storefront foundation layer. The domain provider returns the name of the security domain that is in use to Commerce Connect and Commerce Connect

passes the domain name to the external commerce system on each API call.

## SXA Storefront security roles

Security roles give Sitecore users different access rights to different areas of a site. When you create a new site with Commerce features using the Create a new Experience Accelerator site wizard, Sitecore creates a domain and two security roles, and assigns them to the site.

The domain is created with the same name as the new storefront.

Sitecore creates the following roles:

- Extranet User
- Extranet Customer

These roles do not have any rights or restrictions by default. You can configure the roles to restrict access to sections of the storefront that require authentication.

### Note

Each role belongs to only one domain. You can have two roles that belong to different domains and have the same name. For example, Site1/Extranet User is a different role from Site 2/Extranet User.

## Build a storefront with SXA Storefront

### Set up the SXA Storefront template site

When you install Sitecore Commerce with the SXA Storefront solution, you install functionality that enables you to set up the SXA Storefront template site. However, the site itself has not yet been created in your instance of Sitecore.

The SXA Storefront template site is a full B2C storefront with a range of functionality.

To deploy the full SXA Storefront template site with the Habitat catalog:

- “Create a tenant and site with all commerce features” enabled.

You can then modify and customize the storefront for your own use, or you can use it as a best-practice reference implementation.

Alternatively, you can choose to install the commerce renderings for SXA without the other commerce site features, and then use the renderings to build your own commerce site from scratch.

To install the commerce renderings without installing the full storefront template:

- Create a tenant and site with only the Commerce Storefront Components feature enabled.

#### Note

When you create a new commerce site, the Storefront Mock-up with product images theme is assigned by default to the site. To see what the storefront looks like with a production ready theme, you can [assign the Storefront Branded theme](#) to the site.

## Create a new tenant and site with commerce features

You can create new tenants and sites using wizards in SXA. The SXA Storefront solution extends the wizards to include additional commerce features. In order to include those features in new sites and tenants that you create, you must select the relevant modules when you run the Create a tenant wizard and the Create a new Experience Accelerator site wizard.

This topic describes how to:

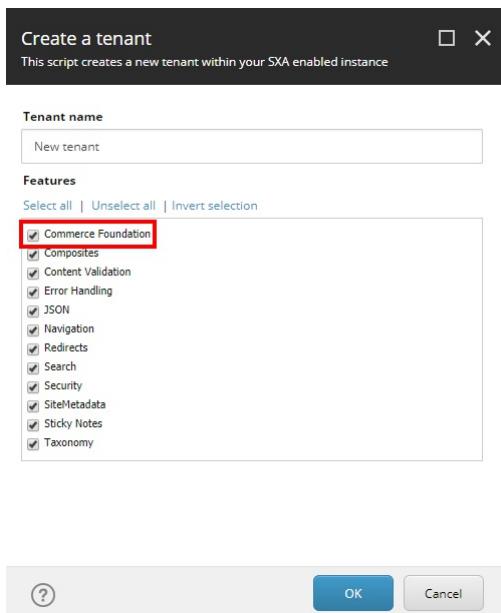
- Create a commerce tenant
- Create a commerce site

### Create a commerce tenant

To create a site with commerce features, you must first create a tenant that includes the Commerce Foundation feature.

To create a commerce tenant:

1. [Create a tenant](#) using the Create a tenant wizard.
2. In the Modules section, select the Commerce Foundation check box together with other features that you want to include.



When you run the wizard with the Commerce Foundation module selected, the tenant that is created contains templates that support commerce storefront sites in addition to the standard SXA templates.

## Create a commerce site

You can use the Create a new Experience Accelerator site wizard to create sites with support for commerce features.

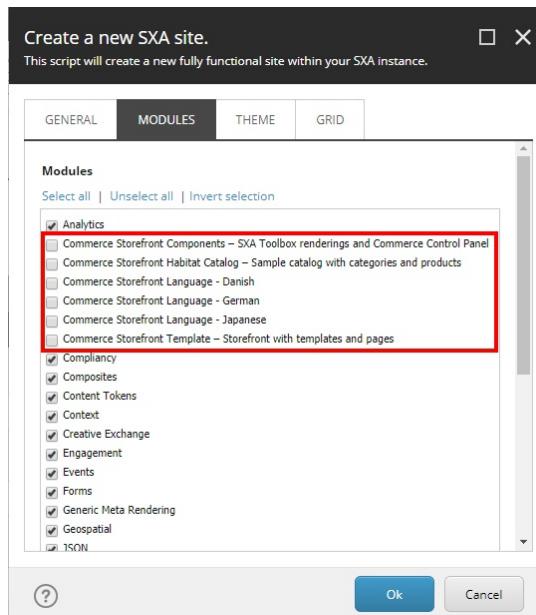
To create a commerce site:

1. In the Content Editor, select the commerce tenant, and [create a site](#) using the Create a new Experience Accelerator site wizard.

### Note

When you create a storefront using SXA Storefront modules, on the General tab, in the Language field, select English (en). From SXA Storefront version 1.0.2, you can select one or more additional languages on the Modules tab. You can also add languages after you create the site.

2. On the Modules tab, select one or more of the Commerce modules, together with other modules that you want to include.



You can select the following Commerce modules:

Module	Comments
Commerce Storefront Components	Installs all Commerce renderings in the SXA Toolbox. You can use the renderings to create commerce sites.

	<p>Also creates a site entry under the Commerce Storefront Control Panel, which is required by the Commerce Foundation layer and Commerce renderings.</p> <p>When you select this module without selecting the Storefront Template feature, only a single empty home page item is created. You can then use the Commerce renderings to build your own storefront pages.</p> <p><b>Note</b></p> <p>To use any part of the SXA Storefront functionality, you must enable this module.</p>
Commerce Storefront Habitat Catalog	<p>Sets the Habitat demonstration catalog as the default catalog for the site. The Habitat catalog items will be available under the Product catalog item for the site (<code>/sitecore/content/[tenant]/[site]/home/Product Catalog</code>), and the Habitat catalog images will be available in the <i>Media</i> folder for the site (<code>/sitecore/content/[tenant]/[site]/Media</code>).</p> <p><b>Note</b></p> <p>Requires the Commerce Storefront Components module.</p>
Commerce Storefront Language - Danish Commerce Storefront Language - German Commerce Storefront Language - Japanese	<p>For each language that you select, Sitecore adds an additional language version and includes translated default content.</p> <p><b>Note</b></p> <p>Requires the Commerce Storefront Template module.</p>
Commerce Storefront Template	<p>Installs the Storefront template site. This provides you with a pre-built storefront with standard pages and functionality using SXA and Commerce renderings, page designs, and partial designs.</p> <p><b>Note</b></p> <p>Requires the Commerce Storefront Components</p>

module.

#### Note

To deploy the full SXA Storefront with the Habitat catalog you must create a tenant and site with all commerce modules enabled.

3. On the Theme tab, select the [themes](#) that you want to be able to apply to your site

If you select the Create new theme check box, a simple, unbranded new theme is created. If you have selected the Commerce Storefront Components feature on the Modules tab, the new theme includes support for the commerce renderings. The support does not include layout and the component styling is minimal. The theme includes all the component [Sass files](#). (Some of the files are empty.) This makes it easier to export the theme using [Creative Exchange](#).

#### Note

If you select the Commerce Storefront Template check box on the Modules tab, the *Storefront Branded*, *Storefront Mock-up with product images*, and *Storefront Mock-up with mock images* themes are installed, regardless of your selection, and the *Storefront Mock-up with product images* theme will be set as the default theme for the site.

The Commerce renderings do not support the Wireframe theme.

4. On the Grid tab, select the bootstrap grid system

#### Note

The Commerce Storefront Template module is only compatible with the Bootstrap grid system. If you select the Commerce Storefront Template check box on the Modules tab, then you must select the Bootstrap check box on the Grid tab.

#### [Security](#)

From SXA Storefront version 1.0.2, when you create a new site with the Create a new Experience Accelerator site wizard, [a security domain and two new extranet user roles](#) are created.

#### [Commerce Engine Configuration](#)

In Commerce Engine, you configure allowed domains with the `Sitecore.Commerce.Plugin.Customers.CustomerPropertiesPolicy` policy as part of the environment. A system administrator needs to update the configuration of the Commerce Engine environment to include the new domains that are created for each site in SXA Storefront. For more information about updating the Commerce Engine configuration, see the Commerce DevOps Guide.

## Configure your storefront

After you [create a site](#) with the SXA Storefront template, you can configure it to transform it into your company's own storefront.

To configure the SXA Storefront:

1. In the *Sitecore/Storefront/Settings/Site Grouping/Storefront* item, in the Host Name field, change the host name to reflect your own storefront:

The image shows two side-by-side screens. On the left is a Sitecore navigation tree. On the right is a configuration interface for the 'Storefront' item under 'Basic' settings.

**Sitecore Navigation Tree:**

- sitecore
  - Sitecore Commerce
  - Content
    - Home
    - Sitecore
      - Storefront
        - Home
        - Media
        - Data
        - Presentation
      - Settings
        - Browser Title
        - Cookie Warning
        - Creative Exchange Storages
        - Datasource Configurations
        - Default links
        - Facets
          - Item Queries
          - Maps Provider
        - Redirects
        - Scopes
      - Site Grouping
        - Storefront
        - Social Media Groups
        - Twitter Apps
    - Layout

**SXA Storefront Configuration:**

  - Quick Info**
  - Basic**
  - TargetHostName [shared]:** (empty)
  - Host Name (use \* as wildcard and | to list more values) [shared]:** (highlighted with a red box)  
SXA Storefront
  - Virtual Folder [shared, standard value]:** (empty)
  - Start Item [shared]:** Storefront/Home
  - Database [shared, standard value]:** web
  - Domain [shared, standard value]:** extranet

2. In the Commerce Control Panel, update the settings. You must update both the [settings installed with Commerce Connect](#), and the settings that are installed with the SXA Storefront.
3. Configure the Commerce renderings:

You can configure all textual content appearing on the storefront per

shop, either [in the Experience Editor](#) or [on the associated content item](#).

### Note

The content items that are associated with the renderings used in the SXA Storefront are located in:

`sitecore/Content/Sitecore/<YourStorefront>/Data/Commerce/<Toolbox category>/<Rendering name>` (for example,

`sitecore/Content/Sitecore/Storefront/Data/Commerce/Account/Registerat`

Configure other aspects that are unique to the specific rendering. You can read how to configure the individual renderings in the following topics that describe a group of renderings in the toolbox.

- Commerce Account renderings
  - Commerce Orders renderings
  - Commerce Cart renderings
  - Commerce Catalog renderings
  - Commerce Checkout renderings
  - Commerce Shared renderings
4. To update the visual design of the storefront, design your own theme.
  5. Configure the content on the storefront pages. For example, insert your own logo image, associate your own images with the Carousel rendering on the *Home* page, and modify text that is not part of the renderings.

## The Commerce Control Panel settings for SXA Storefront

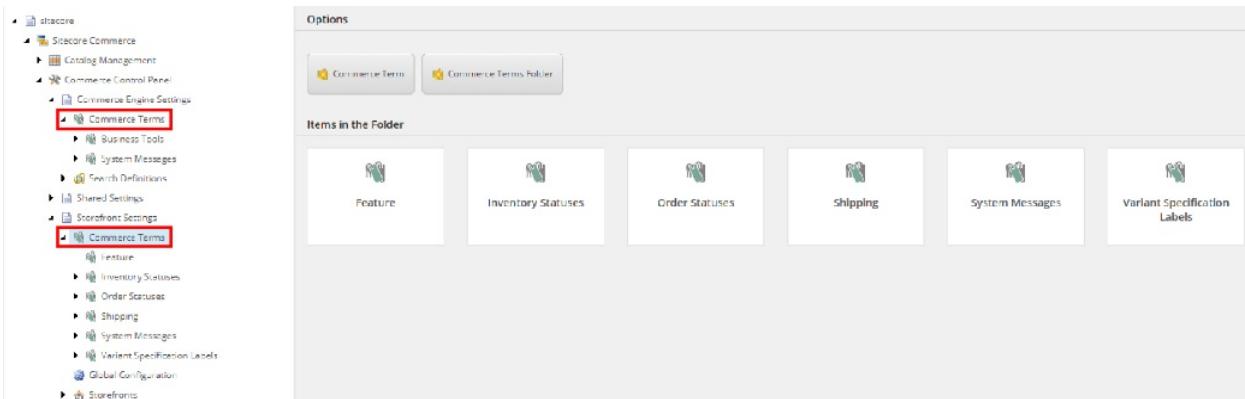
The Commerce Control Panel is installed automatically when you install Commerce Connect. It contains [commerce engine and storefront settings](#) that you must configure. When you install the SXA Storefront, a number of folders with additional configuration items are added to the Commerce Control Panel.

This topic describes:

- The Commerce terms installed with SXA Storefront
- The Global Configuration item
- Storefront-specific settings installed with SXA Storefront

### The Commerce terms installed with SXA Storefront

When you install the SXA Storefront, a *Commerce Terms* folder is installed in the *Storefront Settings* folder (`/sitecore/Commerce/Commerce Control Panel/Storefront Settings/Commerce Terms`). These terms are similar to the terms in the *Commerce Terms* folder that is installed for the commerce engine (`/sitecore/Commerce/Commerce Control Panel/Commerce Engine Settings/Commerce Terms`), but the terms in the *Storefront Settings* folder are specific to your storefronts.



The *Commerce Terms* folder enables you to assign user-friendly terms to various status options that are used by the commerce system, such as inventory and shipping statuses.

In the *Commerce Terms/System Messages* folder, you can edit system messages used on the storefront.

## The Global Configuration item

When you install SXA storefront, the *Global Configuration* item is installed in the *Storefront Settings* folder (`/sitecore/Commerce/Commerce Control Panel/Storefront Settings/Global Configuration`).

You use the *Global Configuration* item to provide configuration information that applies to all storefronts.

You use the Catalog Item Encoding fields to manage encoding values for your storefronts. In some cases, data strings imported from the commerce system can introduce errors in Sitecore. For example, a "/" in a category or subcategory name in a URL would cause the Sitecore routing table to read the string as a product rather than a category, because the string appears to contain an extra path segment.

This issue is avoided by replacing the "/" character with other characters when the URL is generated, and then reverting to "/" when searching for the category. The replacement is made based on an encoding rule in the Catalog Item Encoding field. You can add characters to encode to the list. Each time you add a character to the list, a new empty line appears at the bottom of the list.

The URL Token Delimiter field defines the character used as a separator between the catalog item ID and catalog item name. For example, if the system is configured to use Shop URLs, the separator is placed between the category ID and category display name as well as the product id and product display name.

The Encoded Delimiter field contains the encoding rule for the URL token delimiter.

## Storefront-specific settings installed with SXA Storefront

When you install SXA storefront, four custom configuration items are added to the `/sitecore/Commerce/Commerce Control Panel/Storefront Settings/Storefronts/Storefront` folder:

Setting	Description
<i>Account Management</i>	Determine the maximum number of addresses that can be registered per customer.

Configuration	
<p><i>Catalog Configuration</i></p>	<p>Note</p> <p>From SXA Storefront 1.0.2, the Catalog field and the Start Navigation Category field are deprecated. Use the corresponding fields on the <code>/sitecore/content/[your tenant]/[your site]/Settings/Commerce/Catalog Configuration</code> item to <a href="#">configure the catalog for the storefront</a>.</p> <p>Select the Use Index File for Product Status in Lists field to improve performance when retrieving the product status for lists of products, for example, for the Product List rendering. By default, product statuses are retrieved live from the commerce engine by separate calls.</p> <p>The value in the Default Items Per Page field is used as the default for all categories. To change the setting for a single category, in the <i>Sitecore/Sitecore Commerce/Catalog Management</i> folder, change the value in the Items per page field.</p> <p>The value affects the Product List rendering, and the Product List Header rendering, which allows the customer to select the number of products to show on a category or search result page.</p> <p>In the Gift Card Product ID field, enter the product ID that the commerce system uses for gift cards.</p> <p>The Gift Card Page Link field contains the URL for the gift card product page.</p> <p>In the Shop Pages Root Paths section, the Shop Page Root Path, Category Page Root Path and Product Page Root Path fields must be set to the relative path for the items used when generating links for category and product navigation.</p> <p>Renderings in the Commerce Catalog group that generate links will use the value of the Shop Page Root Path field to generate the URLs.</p> <p>The <a href="#">Metadata meta rendering</a> uses the values configured in the Category Page Root Path and</p>

	Product Page Root Path fields to generate a canonical URL in the head section of each page that contains the rendering. Canonical URLs are used by search engines to improve indexing of categories and products.
<i>Storefront Configuration</i>	Enforce SSL is enabled by default to ensure a private and secure connection between the client browser and the storefront. It is not recommended to disable this feature except during development and testing of the solution.
<i>Email Configuration</i>	In the From Email Address field, enter the email address to use as the sender address for email messages sent by the storefront. The Email Address Validation Regex contains the regular expression used to validate email addresses, for example, for the Login, Registration, and Checkout Billing renderings.

## Configure Catalogs in SXA Storefront

### Note

This topic describes catalog configuration from SXA Storefront 1.0.2. In SXA Storefront 1.0.0 and 1.0.1, you configure the catalog and the categories to display on the storefront in the [Commerce Control Panel](#), on the *Catalog Configuration* item under the respective storefront.

When you create a site using SXA storefront, the wizard creates the */sitecore/content/[your tenant]/[your site]/Home/Catalogs* and */sitecore/content/[your tenant]/[your site]/Settings/Commerce/Catalog Configuration* items. You can use these items to configure the catalogs for the site.

This topic describes how to:

- Configure catalogs for a storefront
- Select the default catalog
- Select the categories to display on the Home page
- Include subcategory results on a category page

### Configure catalogs for a storefront

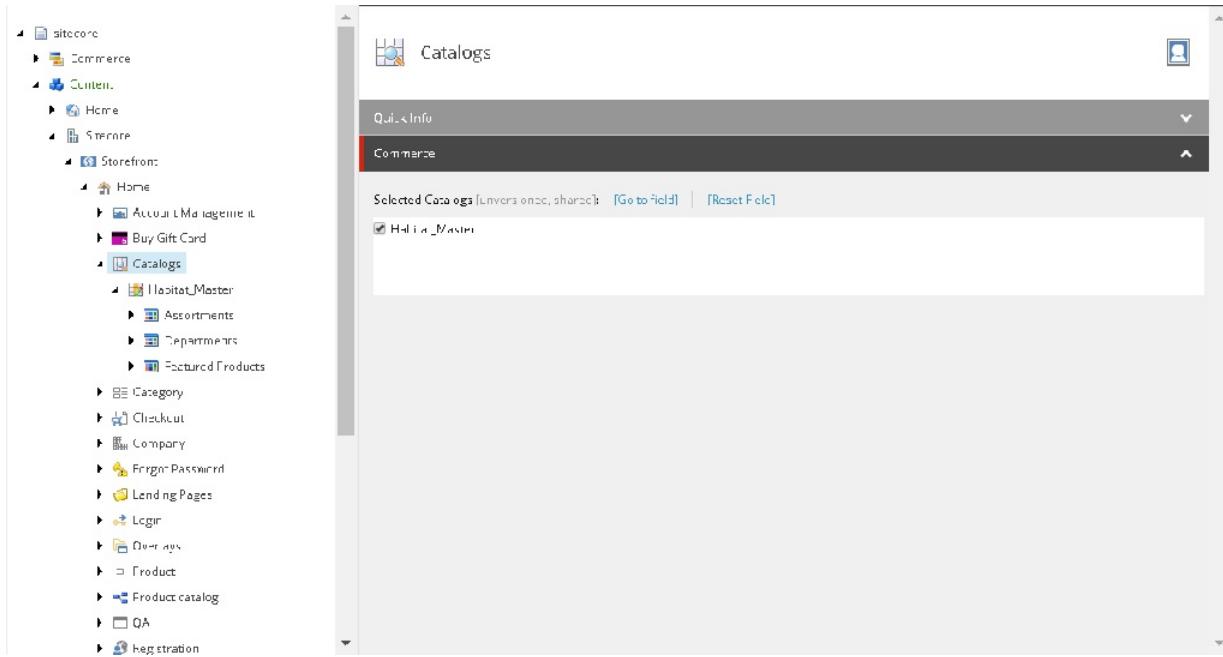
You can configure multiple catalogs for a storefront. Each catalog can only be used by one storefront.

### Note

Multiple sites cannot share the same catalog in Sitecore as the catalog will only be indexed in one location and the other sites will not recognize catalog content outside their own site area. You can use a Postman script to clone existing catalogs if you want the same products on multiple storefronts. For more information about cloning a catalog, see the Commerce DevOps Guide.

To configure catalogs for a storefront:

1. Navigate to the */sitecore/content/[your tenant]/[your site]/Home/Catalogs* item.
2. In the Selected Catalogs field, select one or more catalogs and click Save.



Content for the catalogs starts to appear under the *Catalogs* item.

1. To make the catalog data available and searchable on the storefront, publish and re-index both the master and web indexes.

## Select the default catalog

To select the default catalog:

1. Navigate to the `/sitecore/content/[your tenant]/[your site]/Settings/Commerce/Catalog Configuration` item.
2. In the Catalog field, select the default catalog to use for the storefront.

## Select the categories to display on the storefront

When customers arrive on your storefront, they can use the category navigation bar and subcategory navigation lists to navigate in your catalog. You can specify what is shown in the navigation bar and subcategory lists on your storefront.

To select categories for the storefront:

1. Navigate to the `/sitecore/content/[your tenant]/[your site]/Settings/Commerce/Catalog Configuration` item.
2. In the Start Navigation Category field, select the category that you want to display in the Category Navigation rendering on your site.

When a user navigates your storefront, the contents of the selected folder appears in the Category Navigation and associated Subcategory Navigation renderings.

For example, on the SXA Storefront template site, in the Start Navigation Category field, you select the *Departments* folder is selected:

The screenshot shows the Sitecore Experience Editor interface. On the left is the Sitecore navigation tree. In the center, the 'Catalog Configuration' page is displayed with the 'Commerce' tab selected. Under 'Start Navigation Category', the path 'sitecore/content/Sitecore/Storefront/Home/Catalogs/Habitat\_Master/Departments' is selected. A checkbox labeled 'Include subcategory products - Display products from current category and subcategories' is checked. On the right, the 'Content' ribbon shows a detailed tree view of the 'Departments' folder under 'Habitat\_Master/Assortments'. This tree view is highlighted with a red border. At the bottom of the interface, there are tabs for 'Content Editor', 'Media Library', and 'Workbox'.

On the live storefront, the category navigation bar contains the subcategories from the *Departments* category:

The screenshot shows a live web page from the SXA Storefront template. The top navigation bar includes links for 'Appliances', 'Audio', 'Cameras', 'Computers and Tablets', 'Connected Home', 'Gaming', 'Gift Cards and Gift Wrapping', 'Health, Beauty and Fitness', 'Home Theater', 'Phones', and 'Televisions'. Below this, a prominent red banner features the text 'Now's the time' and a subtext 'With your new watch you'll be able to manage your time and get on track for a healthier life.' To the right of the banner is a large image of a red and black smartwatch. The overall design is clean and modern, utilizing a white background and a red diagonal graphic element.

Include subcategory results on a category page

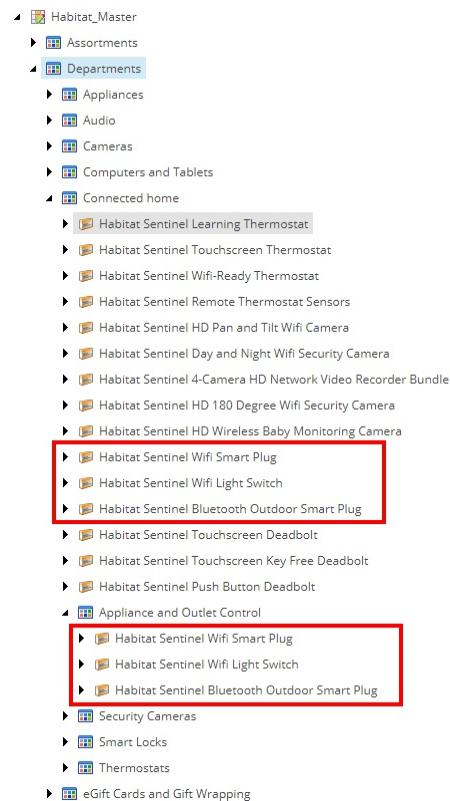
When the user visits a category page, the page can display products in the selected category and all subcategories, or only products from the selected category.

To include products from subcategories on a category page:

- Select the **Include subcategory products** check box. Clear the check box to exclude subcategories from the product list.

## Catalog structure

The Habitat catalog is structured so that products associated with a subcategory such as *Appliance and Outlet Control* are also associated with the higher-level *Connected home* category. With that structure, all products in subcategories of *Connected home* appear when the *Connected home* category is selected.



If you select the **Include subcategory products** check box, you can associate products with subcategories only in the catalog and they will still appear in the higher-level categories on the storefront.

### Note

From Sitecore XC 9.0.2, if a product is associated with multiple categories, it will have a unique ID and path in each location and each instance will be indexed as an individual item. As a consequence, searches on the storefront result in multiple hits for the same product.

## Build a storefront with Commerce renderings for SXA

The Commerce renderings for [Sitecore Experience Accelerator \(SXA\)](#) are commerce-specific renderings that are added to the [SXA toolbox](#) when you create a commerce tenant and site. The Commerce renderings can be used by web development teams to speed up the production of e-commerce storefronts. Using SXA, which separates structure from design, allows front-end developers, creative designers, content authors and developers to work in parallel.

### Note

For best results, familiarize yourself with the functionality of SXA before you begin working with the Commerce renderings.

The Commerce renderings for SXA are specifically tailored for e-commerce and target Sitecore Commerce Engine.

You can use the Commerce renderings to build a storefront based on the prebuilt SXA Storefront or to build a storefront from the bottom up. Building on top of the SXA Storefront gives you a head start, and then you customize the SXA Storefront to make it your own.

If you choose to build your storefront from scratch, you can [build your own page designs](#) using the Commerce renderings in the SXA toolbox.

### Note

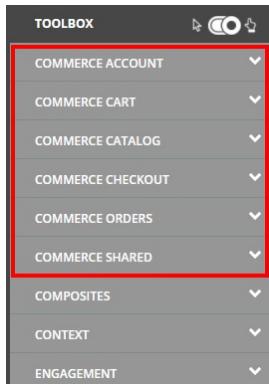
The Commerce renderings are dependent on the Commerce Antiforgery meta rendering. You must include the Antiforgery meta rendering on all storefront pages to ensure that the Commerce renderings behave as expected.

In the SXA Storefront template, the Antiforgery meta rendering is included in the <body> of the Commerce Metadata partial design so that it generates the antiforgery token HTML that is required by the site.

To add a Commerce rendering to a page:

1. In the Experience Editor, open an existing page or partial design, or create a new page or partial design on which you want to include the rendering.
2. In the Toolbox, select the relevant rendering from one of the Commerce

categories:



3. Click and drag the rendering to [add it to the page](#).
4. If relevant, [select or create associated content for the rendering](#).

#### Note

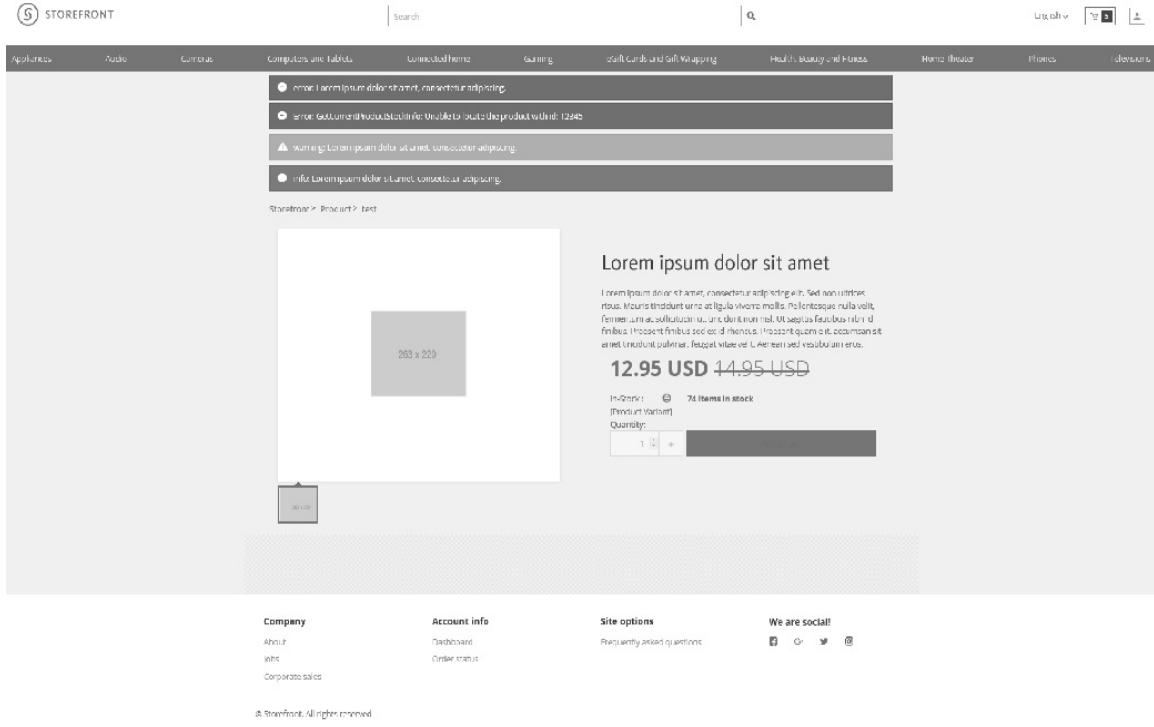
If you are building your site based on the SXA Storefront, content items that you can associate with the renderings have already been created in:

*sitecore/Content/Sitecore/Your Storefront/Data/Commerce/Toolbox category/Rendering name*

*For example,*

*sitecore/Content/Sitecore/Storefront/Data/Commerce/Account/Register*

The selected rendering appears on the page. Some renderings can contain placeholder images and data. For example, a product page looks like this in the Experience Editor, with a *Storefront Mock-up* theme applied:



## Note

The appearance of the renderings will depend on the [theme](#) that is applied to the site.

5. You can edit field labels, button labels, and placeholder text in the rendering:
  - You can [edit some content directly on the page](#) in the Experience Editor.
  - You can edit other content only in the Content Editor. To open the associated content item in the Content Editor, select the rendering and click  , then click Edit the related item.

## Note

In addition to the text strings that are visible in the rendering in the Experience Editor, there can be validation messages that are only displayed on the live site. The validation messages can be edited on the associated content item in the Content Editor.

Other configuration options are described individually for each rendering, in the following topics:

- Commerce Account renderings
- Commerce Cart renderings
- Commerce Catalog renderings
- Commerce Checkout renderings
- Commerce Orders renderings
- Commerce Shared renderings

#### Note

In SXA Storefront 1.0, components that contain forms for data input are implemented using standard .NET MVC and AJAX form techniques. Therefore, on predefined forms you can only customize the labels on input fields and buttons. A future version of SXA Storefront will contain forms implemented with the Sitecore Forms module that is included with Sitecore Experience Platform 9.

## Add a language version to an existing storefront site

When you use the Create a new Experience Accelerator site wizard to create a storefront site with the Commerce Storefront Template, you can include one or more pre-translated languages in addition to English.

You can also add a new language to an already existing site that was created with the Commerce Storefront Template. When you do this, Sitecore adds a language version of each item from the template, and includes translated default content.

To add a language version to an already existing site:

1. In the Content Editor, right-click the site and then click Scripts, Add Site Module.

### Note

Do not click Scripts, Add a Language to add a language to a site created with the SXA Storefront Template. Only use this method to add a language version to an SXA site that was created *without* Commerce Storefront modules. It will not include the translated default content that comes with the Commerce Storefront Template.

2. In the dialog box that appears, select the language or languages that you want to add and click OK.

The wizard adds the selected languages to the current storefront site.

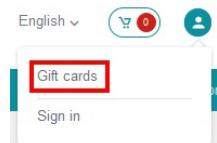
### Note

Any changes or additions that you make to the site have to be replicated in each language version. For example, if you add a component to the English version of a page, no change is made automatically to the other versions of the page - you have to add the component to each version. Default translations are included with the component, but if you make changes to the English text, you have to translate the changes.

## Configure the Buy Gift Card page

The SXA Storefront includes a Buy Gift Card page that customers can access from the Top Bar Links menu and from the gift card category page.

### Top Bar Links menu



### Gift card category page

A screenshot of a product category page titled 'eGift Cards and Gift Wrapping'. The left sidebar shows filters for 'Gift cards and gift wrapping', 'Gift Cards', 'Gift Wrapping', and 'Brand' (Habitat Gift Wrapping). The main content area shows two products: 'Habitat Gift Wrapping' (4.00 USD) and 'Habitat Gift Cards' (starting from 25.00 USD). The 'Habitat Gift Cards' product is highlighted with a red rectangular box.

There are no SXA Toolbox renderings specifically for the Buy Gift Card page; the page is composed of renderings that are also used on other product pages. You must specify which product in the catalog you want to treat as a gift-card product.

To configure a gift-card product in the Commerce Control Panel:

1. In the Content Editor, navigate to the *Catalog Configuration* item in the */sitecore/Commerce /Commerce Control Panel/Storefront Settings/Storefronts/Storefront* folder.
2. In the Gift Card section, configure the following fields:
  - Gift Card Product ID – specify the product ID from the catalog for the gift-card product. When the customer selects this product, Sitecore redirects the customer to the Buy Gift Card page.

### Note

Gift cards with different values can be variants of the same product.

- Gift Card Page Link – specify the URL for the Buy Gift Card page. The default is `/buygiftcard`.

## Commerce renderings for SXA

### Commerce Account renderings

You can add Commerce Account renderings to your storefront to enable account and profile management.

This topic describes the following renderings in the Commerce Account group in the SXA toolbox:

- Registration
- Login
- Address List
- Address Editor
- Profile View
- Profile Editor
- Change Password
- Forgot Password

#### Registration

You use the Registration rendering to insert a registration form on your website. If a customer visits your website and does not already have an account, they use the registration form to create a new account. The system registers the new account and then automatically logs the customer in for the current session. When the account is created, the *User Account Created* goal is triggered in the Experience Platform.

In the Experience Editor, with the basic custom theme applied, it looks like this:

A screenshot of a registration form in the Experience Editor. The form consists of three input fields: 'Email address' (disabled), 'Password' (disabled), and 'Repeat password' (disabled). Below the fields are two buttons: 'Create new account' (disabled) and 'Continue as guest' (disabled).

In the Experience Editor, fields and buttons in the rendering are disabled, and the registration form cannot be submitted.

### *The Registration rendering on the live storefront*

While the page is loading on the storefront, the registration form is disabled. Once the page is fully loaded, the form is enabled.

### *Field validation*

The customer is required to fill in all fields on the registration form. When the customer submits the form, Sitecore validates the following:

- There is content in all fields.
- The email address format is correct.

#### **Note**

You can edit the regular expression used for email format validation in the Commerce Control Panel, in the *Email Configuration* item ([/sitecore/Commerce/Commerce Control Panel/Storefront/Settings/Storefronts/Storefront/Email Configuration](/sitecore/Commerce/Commerce%20Control%20Panel/Storefront/Settings/Storefronts/Storefront/Email%20Configuration))

- The password is at least 6 characters long.
- The content of the Password and Confirm Password fields is identical.

If any of the validations fail, an error message is displayed for each of the empty or incorrect fields, for example:

The screenshot shows a registration form with four fields: Email address\*, Password\*, Repeat password\*, and two buttons at the bottom: Continue as guest and Create new account. The 'Email address\*' field contains 'test' and has a red border. An error message 'Please enter a valid email address.' is displayed below it. The 'Password\*' field contains '\*\*\*\*' and has a red border. An error message 'The password must be at least 6 characters long.' is displayed below it. The 'Repeat password\*' field contains '\*\*\*\*' and has a red border. An error message 'The password and confirmation password do not match.' is displayed below it. The 'Create new account' button is highlighted in blue.

Email address*	test
Please enter a valid email address.	
Password*	****
The password must be at least 6 characters long.	
Repeat password*	****
The password and confirmation password do not match.	
Continue as guest	Create new account

### *Registration form submission*

When the customer clicks Create new account, the system validates the fields. The button is disabled to avoid multiple clicks and the button label changes to Registering.... If there are no validation errors, the form is submitted to the server.

### *Valid registration information*

If the email address provided for registration does not exist in the system, the

customer is added to the system and logged in automatically. As part of the login process, the customer is redirected to the Account Management page.

#### [Invalid registration information](#)

If the email address provided for registration already exists in the system, and if the Message Summary component is on the page, the Message Summary displays an error message.

#### Note

Some messages that appear in the message summary are system messages, which you can configure in the Commerce Control Panel, in the *Storefront Settings/Commerce Terms/System Messages* folder.

### Login

You use the Login rendering to insert a login form on your website. When a customer who has registered an account returns to the site, they use the login form to gain access to their account.

The SXA toolbox contains two Login renderings:

- A Login rendering in the Security group – used on a webpage to authenticate contacts or customers.
- A Login rendering in the Commerce Account group – also used on a webpage to authenticate contacts or customers but it has additional functionality. You must use this rendering when you make a website that is integrated with the Commerce Engine to ensure that the login is authenticated in the commerce system.

In the Experience Editor, fields and buttons in the rendering remain disabled, and the Login form cannot be submitted. With the basic custom theme applied, it looks like this:

A screenshot of a login form on a website. The form consists of three main input fields: 'Email address\*' with placeholder 'Enter email address', 'Password\*' with placeholder 'Enter password', and a 'Sign in' button below them. All fields and the button are currently disabled, appearing in a light gray color.

#### [The Login rendering on the live storefront](#)

While the page is loading on the storefront, the Login form is disabled. Once the page is fully loaded, the form is enabled.

#### [Field validation](#)

The customer is required to fill in both the Email Address and Password fields on the form. When the customer submits the form, Sitecore validates the following:

- There is content in all fields
- The email address format is correct.

#### Note

You can edit the regular expression used for email format validation in the Commerce Control Panel, in the *Email Configuration* item (`/sitecore/Commerce/Commerce Control Panel/Storefront/Settings/Storefronts/Storefront/Email Configuration`)

If any of the validations fail, an error message is displayed for each of the empty or incorrect fields, for example:

A screenshot of a login form. It has two fields: 'Email address\*' and 'Password\*'. The 'Email address\*' field contains 'Test' and has a red border around its input area, with the error message 'Please enter a valid email address.' below it. The 'Password\*' field contains 'Enter password' and also has a red border around its input area, with the error message 'The password is required.' below it. A teal 'Sign in' button is at the bottom.

#### *Login form submission*

When the customer clicks Sign in, the system validates the fields. The button is disabled to avoid multiple clicks and the button label changes to Authenticating.... If there are no validation errors, the form is submitted to the server.

#### *Valid login information*

If the email address and password match with a valid account in the system, the customer is redirected to a specified URL. If the Top Bar Links rendering is used on the site, the Sign in link is changed to Sign out.

#### *Invalid login information*

If the email address and password do not match with a valid account in the system, a message is sent to the Message Context, which serves as a

communication broker between components. If the Message Summary component is on the page, it will pick up the message from the Messaging Service and display it. It might look like this, for example:

The screenshot shows a 'Sign in' page with a red box highlighting a message at the top: 'Info: The Email Address or Password provided is incorrect.' Below this, the URL 'Home > Sign in' is visible. The main form has fields for 'Email address\*' containing 'invaliduser@test.com' and 'Password\*' containing '\*\*\*\*\*'. A teal 'Sign in' button is at the bottom.

## Address List

You use the Address List rendering to display the customer's shipping and billing addresses. For example, this rendering is used on the Account Management page in the SXA Storefront template.

When you add the rendering in the Experience Editor with the *Storefront Mock-up* theme applied, it is displayed with placeholder data:

## Address book

[No text in field]

Lorem ipsum dolor sit  
Lorem ipsum dolor sit  
consectetur adipiscing elit  
Nunc  
Amet, Proin

✓ Primary

Lorem ipsum dolor sit  
Lorem ipsum dolor sit  
consectetur adipiscing elit  
Nunc  
Amet, Proin

Lorem ipsum dolor sit  
Lorem ipsum dolor sit  
consectetur adipiscing elit  
Nunc  
Amet, Proin

Lorem ipsum dolor sit  
Lorem ipsum dolor sit  
consectetur adipiscing elit  
Nunc  
Amet, Proin

Lorem ipsum dolor sit  
Lorem ipsum dolor sit  
consectetur adipiscing elit  
Nunc  
Amet, Proin

---

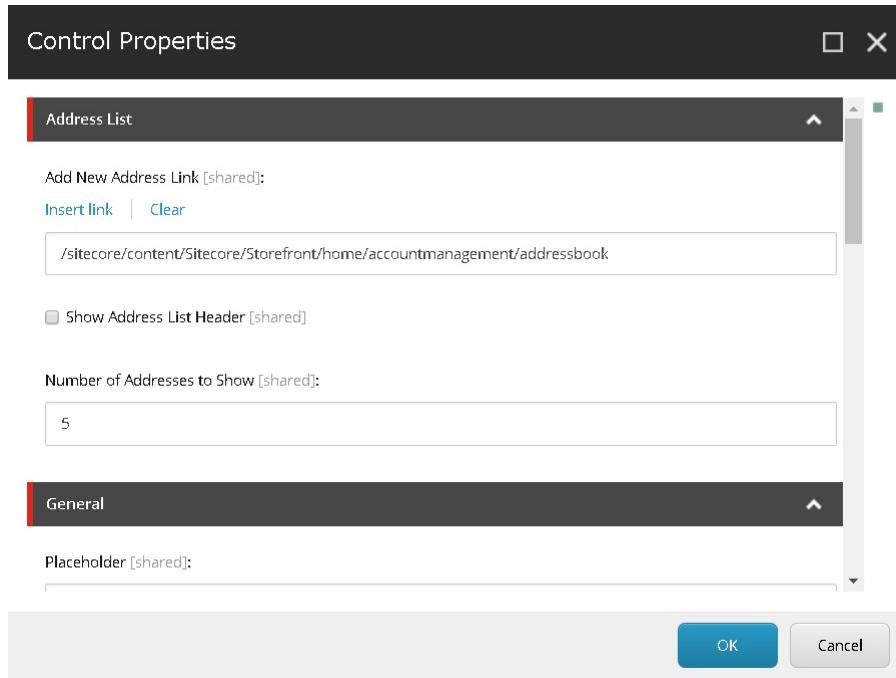
+Add a new address

## Configuration

In the associated content item, in the Number of Addresses to Show field, you can configure the maximum number of addresses displayed in the list.

In the Show Address List Header field, you can configure whether to show the header or not.

In the Add New Address Link field you can configure the link to redirect to, when an address or Add a new address link is clicked.



### *The Address List rendering on the live storefront*

On the live storefront, the Address List rendering displays addresses that have been registered for the customer.

#### Note

In the Account Management Configuration item (in </sitecore/Commerce/Commerce Control Panel/Storefront> [Settings/Storefronts/Storefront/Account Management Configuration](#)), you can configure the maximum number of addresses that a customer can register.

If there are no addresses, a message is displayed to indicate that the address book is empty.

When the customer clicks an address in the list or clicks the Add a new address link, the Address Editor page is displayed.

### Address Editor

The Address Editor rendering enables the customer to add a new address, edit an existing address, or delete an address from the address book.

With a basic custom theme applied, it looks like this:

My saved addresses

Name\* Name

Country/Region \* Country/Region

Address\* Address

City\* City

State/Province State/Province

Zip code\* Zip code

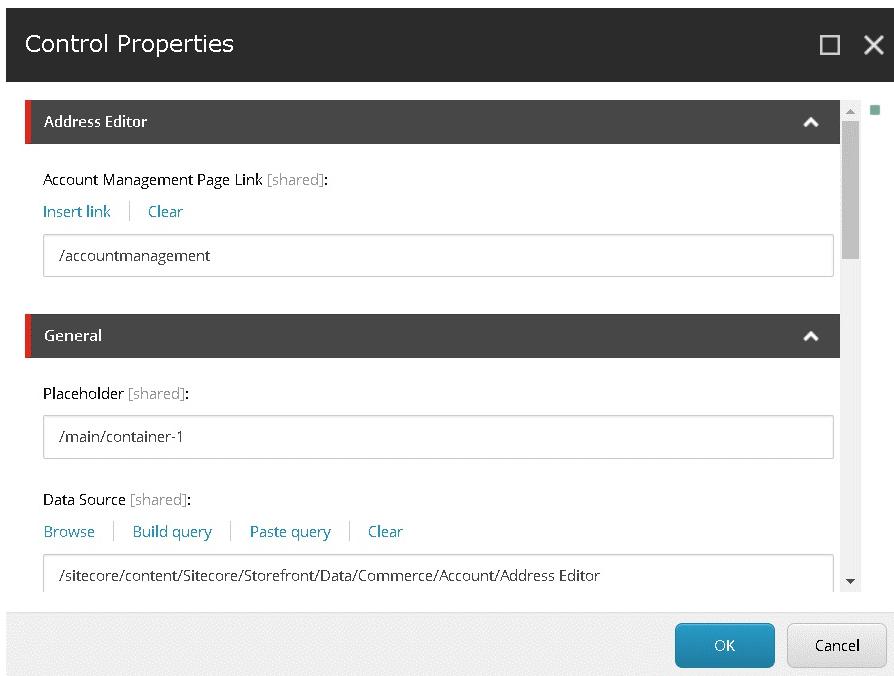
Set as primary address

Save changes Delete Cancel Back to my account

In the Experience Editor, fields and buttons in the rendering are disabled.

### Configuration

In the Control Properties dialog for the rendering, in the Account Management Page Link field, you can configure the link to redirect to after the customer has clicked either Cancel or Save changes.



### The Address Editor rendering on the live storefront

The Set as primary address checkbox is automatically selected for the first address that is entered. If another address is subsequently marked as the primary one, the checkmark is removed from the first address. Only one address can be the primary address.

### Profile View

You use the Profile View rendering to display the email address for the current customer account. In the Experience Editor, the rendering contains

placeholder data for the email address. When you insert the rendering in the Experience Editor with a basic custom theme applied, it looks like this:

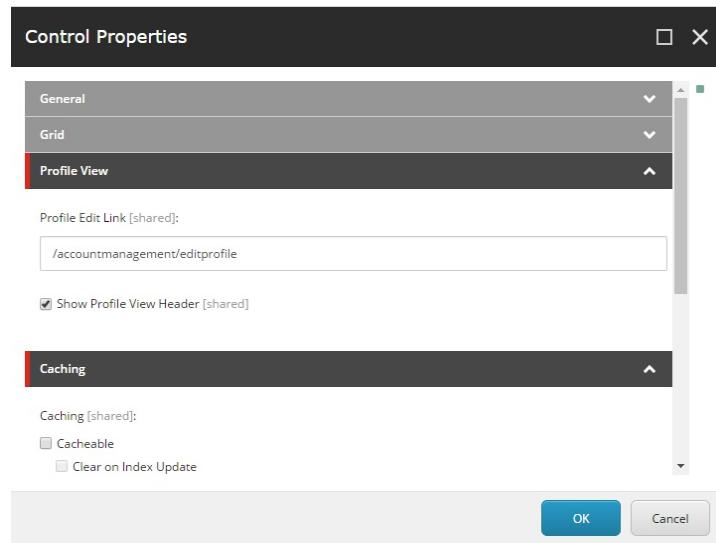


The email address and the edit link are both links to the same URL. This will typically be a page with the Profile Editor rendering, where the customer can edit profile information.

#### *Configuration*

In the Control Properties dialog for the rendering, you can edit the link used for the email address and the Edit link. This will typically be to a page with the Profile Editor rendering.

You can also configure whether the rendering displays the Account profile header.



## Profile Editor

The Profile Editor rendering enables the customer to edit their profile, which consists of the following information:

- First name
- Last name
- Telephone number
- Email address

First name\*  
Last name\*  
Email address\*  
Repeat email address\*  
Telephone number\*

Discard changes      Save changes      Back to account management

When the customer clicks Save Changes, the system validates that the contents of the Email address and Repeat email address fields match. The system also validates that the email address format conforms to the email address regular expression in the Commerce Control Panel.

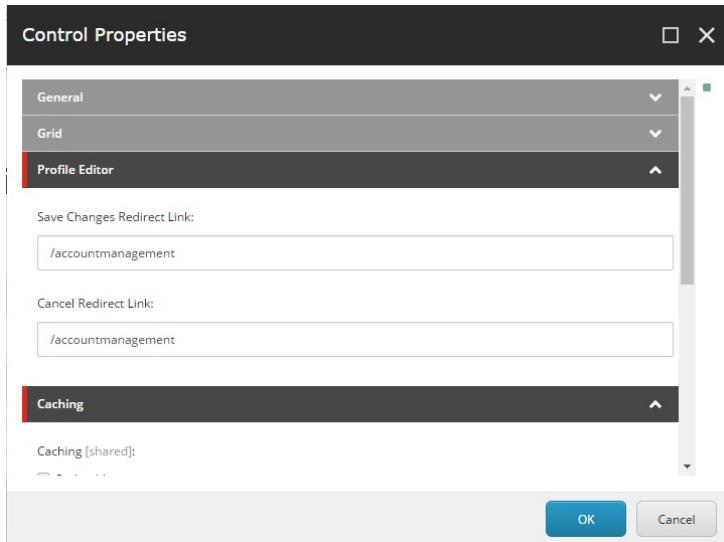
When the content passes validation:

- The Sitecore user (.NET membership provider) is updated with the data.
- The contact information in xDB is updated.
- The corresponding user profile in the Commerce Engine is updated.

The system redirects the customer to a different URL or displays a message confirming that the changes have been saved.

#### *Configuration*

In the Control Properties dialog for the rendering, you can configure a link to redirect the customer after they click Save changes. You can also configure a link to redirect the customer after they click Cancel.



## Change Password

The Change Password rendering looks like this in the Experience Editor:

Current password\*

New password\*

Repeat password\*

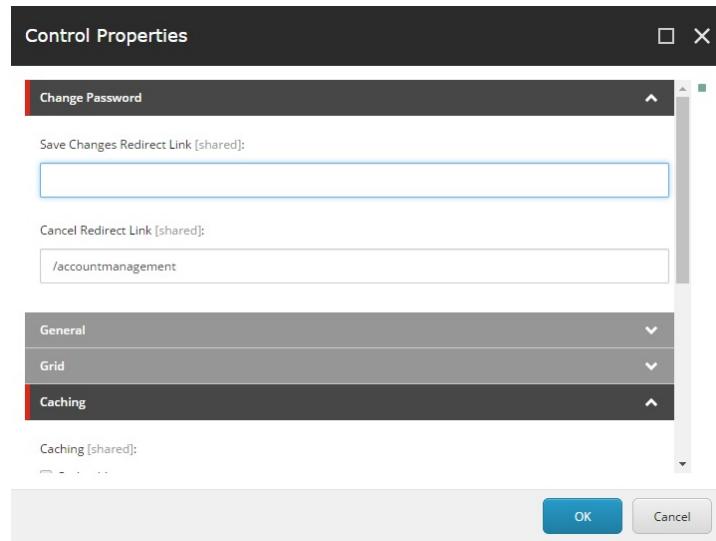
Cancel      Save changes

When the customer clicks Save Changes, the system validates that the content of the Current Password is correct, and that the contents of the New Password and Repeat Password fields match.

When the content passes validation, the system updates the password and redirects the customer to a new page or displays a message confirming that the password has been changed.

#### *Configuration*

In the Control Properties dialog for the rendering, you can configure a link to redirect the customer after they click Save Changes. You can also configure a link to redirect the customer after they click Cancel.



#### *Forgot Password*

The Forgot Password rendering looks like this in the Experience Editor:

Email address\*

**Email Sent**  
A confirmation email has been sent to your primary email address with additional instructions. Follow the steps provided to complete the password reset process.  
Email address: ...

When the customer submits an email address, the system checks for the email in the system.

If the email is recognized, a message is displayed and the system sends an email to the email address, including a random password and a link to the Change Password page on the website.

If the email address is not recognized, nothing happens. For security reasons, no message is displayed.

#### *Configuration*

In the associated data item, you can configure the email message that is sent to the customer when they report a forgotten password.

## Commerce Cart renderings

You can add the Commerce Cart renderings to your storefront so that a contact can add items to their shopping cart when they browse your storefront.

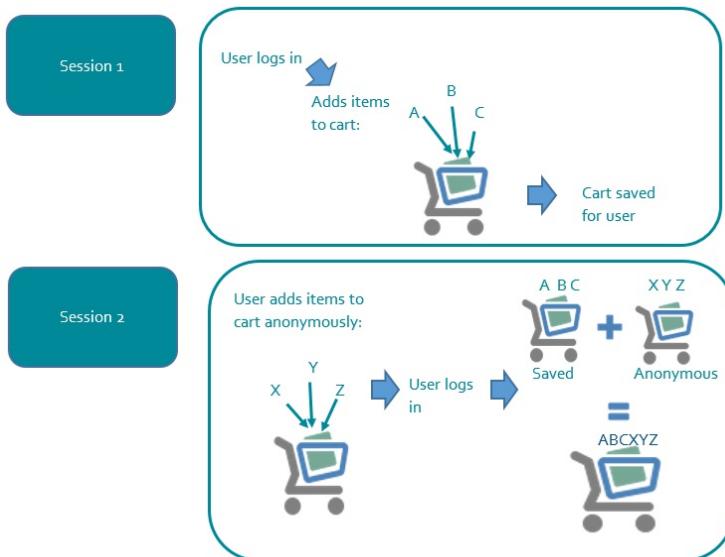
Example product page on the live storefront

The screenshot shows a product page for a 'Habitat Dwell 2-Speed Hand Blender'. The product image is a red hand blender. The product details box contains the product name, price (39.99 USD), a 'Checkout' button, and a 'View cart' button. Below the product details is a 'Minicart' section showing the same item with a price of 39.99 USD. At the bottom of the page is a footer with links for Company, Account Info, Site options, and We are social!.

Example shopping cart page

The screenshot shows a shopping cart page. The 'Product details' section displays a refrigerator with the model 'Habitat Viva 4-Door 22.0 Cubic Foot Refrigerator w/ Ice Maker and Touchscreen (Stainless)' and a price of 4,899.99 USD. The 'Discount code' section has an input field 'Enter discount code' and a button 'Add discount code'. The 'Summary' section shows a breakdown of the total cost: Sub total (4,899.99 USD), Total discount (15.00 USD), Shipping (15.00 USD), Taxes (490.00 USD), and a final Total of 5,389.99 USD. At the bottom are buttons for 'Continue shopping great products' and 'Checkout'.

Customers can browse the storefront and add items to the cart anonymously. If the customer adds items to the cart while browsing the site anonymously, those items remain in the cart after a successful login. If the customer adds products to their cart while logged in, the products remain in the cart after the session ends. Later, if the customer adds items to the cart anonymously and then logs in, the saved cart is merged with the anonymous cart.



This topic describes the following renderings in the Commerce Cart group in the SXA Toolbox:

- Add To Cart
- Minicart
- Shopping Cart Lines
- Promotion Codes
- Shopping Cart Total

## Add To Cart

You use the Add To Cart rendering on product detail pages to enable the customer to add the current product to the shopping cart.

When you add the rendering to a page in the Experience Editor with a basic custom theme applied, it looks like this:

Quantity:  
1  
Add to cart

When the customer clicks Add to cart, the system validates the Quantity field. If the field is empty, a message is displayed with the default text *This field is required*. If the field contains a value that is not a positive integer, a message is displayed with the default text *Please enter a value greater than or equal to 1*.

If the Quantity field passes validation, the system adds a cart line to the shopping cart with the specified quantity, the product ID of the current product, and the catalog and variant ID as needed.

## Minicart

You use the Minicart rendering on catalog and product pages to display a summary of the contents of the shopping cart.

When you add the Minicart rendering to a page in the Experience Editor with the *Storefront Mock-up* theme applied, it looks like this:

English ▾ 5 Shopping Cart Link

Minicart rendering

72 x 72	Lore ipsum dolor sit amet, id dicant	Quantity 1	0.00 USD
72 x 72	Lore ipsum dolor sit amet, id dicant	Quantity 1	0.00 USD
72 x 72	Lore ipsum dolor sit amet, id dicant	Quantity 1	0.00 USD
72 x 72	Lore ipsum dolor sit amet, id dicant	Quantity 1	0.00 USD
72 x 72	Lore ipsum dolor sit amet, id dicant	Quantity 1	0.00 USD
Total		0.00 USD	

On the live storefront, the shopping cart link is visible at all times. When the customer hovers the mouse over the shopping cart link or clicks the link, the Minicart rendering appears. If there are products in the shopping cart, the shopping cart lines and total are displayed.

The customer can click a line to see product details, or click the delete icon to remove a line from the cart.

If the customer clicks View Cart, the Shopping Cart page appears. If the customer clicks Checkout, the first checkout step appears.

## Shopping Cart Lines

You can add the Shopping Cart Lines rendering to a storefront to display the content of the shopping cart with a line for each item. Each line contains the product image, the product name with a link to the product page, the variant, the unit price, the quantity, and the line total, as well as an icon to delete the line. When you add the rendering to a page in the Experience Editor, it looks as follows:

Product details	Unit price	Quantity	Total
 Lorem ipsum dolor sit amet, id dicant Color: Soleat Size: dolor Style: sit Gift card amount: \$25	0.00 USD Discount: mediocritatem no mei(25%)	<input type="button" value="-"/> <input checked="" type="button" value="1"/> <input type="button" value="+"/>	<b>0.00 USD</b> 0.00 USD 
 Lorem ipsum dolor sit amet, id dicant Color: Soleat Size: dolor Style: sit Gift card amount: \$25	0.00 USD Discount: mediocritatem no mei(25%)	<input type="button" value="-"/> <input checked="" type="button" value="1"/> <input type="button" value="+"/>	<b>0.00 USD</b> 0.00 USD 
 Lorem ipsum dolor sit amet, id dicant Color: Soleat Size: dolor Style: sit Gift card amount: \$25	0.00 USD Discount: mediocritatem no mei(25%)	<input type="button" value="-"/> <input checked="" type="button" value="1"/> <input type="button" value="+"/>	<b>0.00 USD</b> 0.00 USD 
 Lorem ipsum dolor sit amet, id dicant Color: Soleat Size: dolor Style: sit Gift card amount: \$25	0.00 USD Discount: mediocritatem no mei(25%)	<input type="button" value="-"/> <input checked="" type="button" value="1"/> <input type="button" value="+"/>	<b>0.00 USD</b> 0.00 USD 
 Lorem ipsum dolor sit amet, id dicant Color: Soleat Size: dolor Style: sit Gift card amount: \$25	0.00 USD Discount: mediocritatem no mei(25%)	<input type="button" value="-"/> <input checked="" type="button" value="1"/> <input type="button" value="+"/>	<b>0.00 USD</b> 0.00 USD 

If any type of discount applies to a line, the type of discount appears under the unit price, and the discount amount appears under the line total.

## Promotion Codes

You can use the Promotion Codes rendering on the Shopping Cart page to enable the customer to enter discount codes for their order. The customer can add a discount code in the Discount code field, and then click Add discount code. The system creates a new line in the rendering for each discount code that the customer enters. Each line contains a Delete icon, so the customer can remove the discount code.

In the Experience Editor, with the *Storefront Mock-up* theme applied, it looks as follows:

Do you have a discount code?

## Shopping Cart Total

You can use the Shopping Cart Total rendering on the Shopping Cart page to summarize the total amounts for the shopping cart contents. In the Experience Editor, with the *Storefront Mock-up* theme applied, it looks as follows:

Summary	
Sub total	<b>0.00 USD</b>
Total discount	<b>0.00 USD</b>
Shipping	<b>0.00 USD</b>
Taxes	<b>0.00 USD</b>
<hr/>	
Total	<b>0.00 USD</b>

If the customer changes something in the shopping cart lines, the totals are automatically updated.

## Commerce Catalog renderings

SXA Storefront includes a number of renderings that you can use together to build a catalog on your storefront.

The following screenshots show examples of a product page and a category page from the SXA Storefront. The red boxes indicate the Commerce Catalog renderings used to build the pages.

### Product detail page example

The screenshot shows a product detail page for a 'Habitat Dwell Bagless Upright Vacuum'. The page is structured with several red boxes highlighting different Commerce Catalog renderings:

- Product Images:** A large image of the vacuum cleaner and a smaller thumbnail below it, both enclosed in a red box.
- Product Information:** A box containing the product name 'Habitat Dwell Bagless Upright Vacuum' and a detailed description of its features.
- Product Inventory:** A box showing the stock status 'In Stock' with '10000 items in stock'.
- Product Price:** A box displaying the price '69.99 USD'.
- Product Variants:** A box showing color options ('Red') and quantity selection ('1' to '+').
- Add to cart:** A button labeled 'Add to cart'.
- Related Product:** A section showing four related products: 'Studio X Over the Ear Wired ...', 'Mira 17.3" Laptop 4GB Me...', 'Habitat Ellipse 65" Curved U...', and 'Habitat Republic 84GB 4G LTE ...'. This section is also enclosed in a red box.
- Promoted Products:** A section showing four promoted products: 'Studio X Over the Ear Wired ...', 'Mira 17.3" Laptop 4GB Me...', 'Habitat Ellipse 65" Curved U...', and 'Habitat Republic 84GB 4G LTE ...'. This section is also enclosed in a red box.

### Category page example

**Product Facets**

**Product List**

**Product List Pager**

This topic describes the following renderings, which are located in the Commerce Catalog group in the SXA Toolbox:

- Catalog Item Container
- Category Navigation
- Subcategory Navigation
- Product Facets
- Product Images
- Product Information
- Product Inventory
- Product List
- Product List Items per Page

- Product List Page Info
- Product List Sorting
- Product List Pager
- Product Price
- Product Variants
- Promoted Products
- Search Initiated, Visited Category Page, and Visited Product Page

## Catalog Item Container

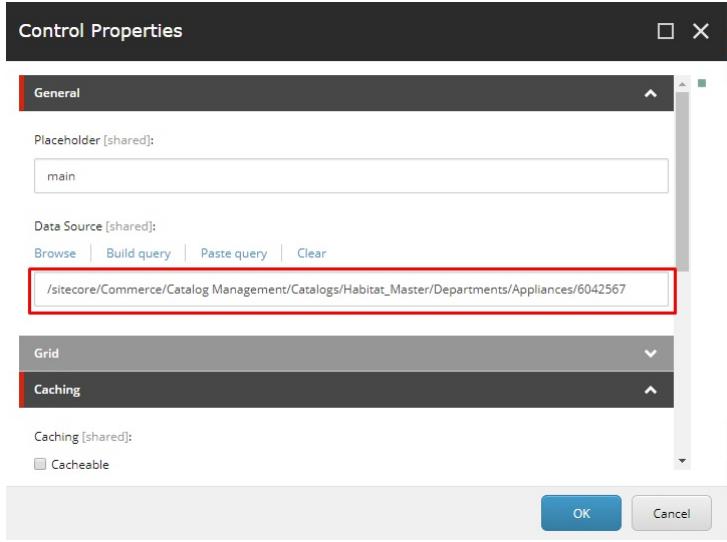
You can use the Catalog Item Container rendering on a landing page to set a product item as the data source for a group of catalog renderings. All the catalog renderings added inside the container rendering will use the product you select as the product context. This allows the renderings to display product information on a page where there is otherwise no product context available.

When you add this rendering to a placeholder, it does not display anything. The rendering contains a placeholder where you can drop other renderings.



### *Setting the data source*

In the Control Properties dialog box for the rendering, you can set the data source to a product item:



After you set the product item and add catalog renderings inside the container rendering, product information for the selected product is displayed.

Catalog renderings added inside the container rendering have the same appearance as if they were added directly to the page, so the landing page looks the same as the product detail page for the selected product.

## Category Navigation

You can place the Category Navigation rendering on a page to create a Navigation bar that displays the top-level categories from your product catalog.

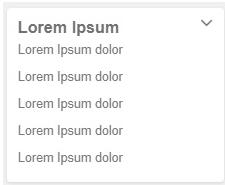


The customer can navigate to a category page by clicking a category in the Navigation bar.

The displayed categories are based on the content of the search index.

## Subcategory Navigation

You can use the Subcategory Navigation rendering to display a list of the subcategories of the selected top-level category. In the Experience Editor, if no top-level category is selected, the rendering is displayed with placeholder text.



## Product Facets

You can add the Product Facets rendering to the category page to display a list of available facets for the currently selected category. If no category is selected, then the rendering is displayed with placeholder text.



The Product Facets rendering is also used on the standard storefront Search page.

### *Filtering a list on the live storefront*

When the customer clicks a facet value in a facet list, the facet list is filtered to only show the selected facet value, and the product list is filtered based on the selected facet value.



It is possible to narrow a filter by selecting multiple facets at once. If the customer clicks a selected facet value, the facet is removed from the search filter.

## Product Images

You can add the Product Images rendering to a page in the Experience Editor

to display thumbnail images for each image related to the selected product. One thumbnail is selected, and a larger image of the selected thumbnail is displayed. If no product is selected, placeholder images are displayed.



## Product Information

You can add the Product Information rendering to a page in the Experience Editor to display the product title, description, and item number.

**Product Information**

Alternatively, you can [use variants of standard SXA renderings](#) to display product information. This allows you to control the placement of the product name, description, and item number independently of each other.

## Product Inventory

You can add the Product Inventory rendering to a page in the Experience Editor to display the current stock status and number of items in stock as [stock status] / [stock count]. When no product is selected, placeholder data is displayed.

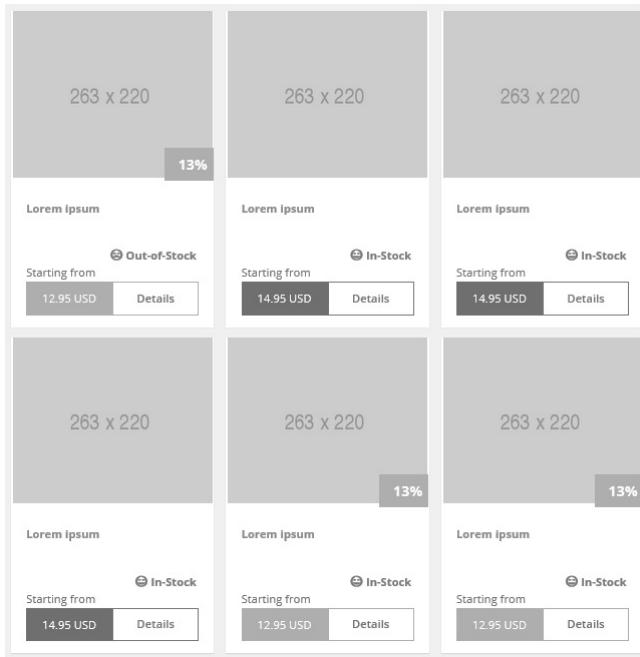
In-Stock :  74 Items in stock

If the product is out of stock, the Add to cart button is disabled.

## Product List

The Product List rendering displays a list of products with the product image and price, and a link to the product page. If you place the Product List rendering on a category page, it displays the products for the selected

category. If no category is selected, placeholder images and data are displayed.



The Product List is also used on the Search page to display search results, which can be a combination of products and categories.

You can place the Product List Items per Page, Product List Page Info, and Product List Sorting renderings above the product list to enable the customer to configure the order of the products in the list and the number of products displayed per page.

If you prefer to display all products in the list on the same page, you can [configure the list to use lazy loading](#).

When a product has variants of different prices, the lowest available price is displayed, prefixed with a text, for example, *Starting from*. If no variants are defined, or if all variants have the same price, the price is displayed without the text.

If a product is on sale at a discounted price, an overlay banner appears overlapping the product picture and displays the amount of the discount as a percentage of the original price. The price that appears in the product list is the discounted price.

The current stock status of the product is displayed.

The customer can navigate to the product details page by clicking the product image or the Details link.

## Product List Items per Page

If you place the Product List Items per Page rendering above the Product List rendering on category and search pages, the customer can select the number of items displayed per page.



### Default items per page

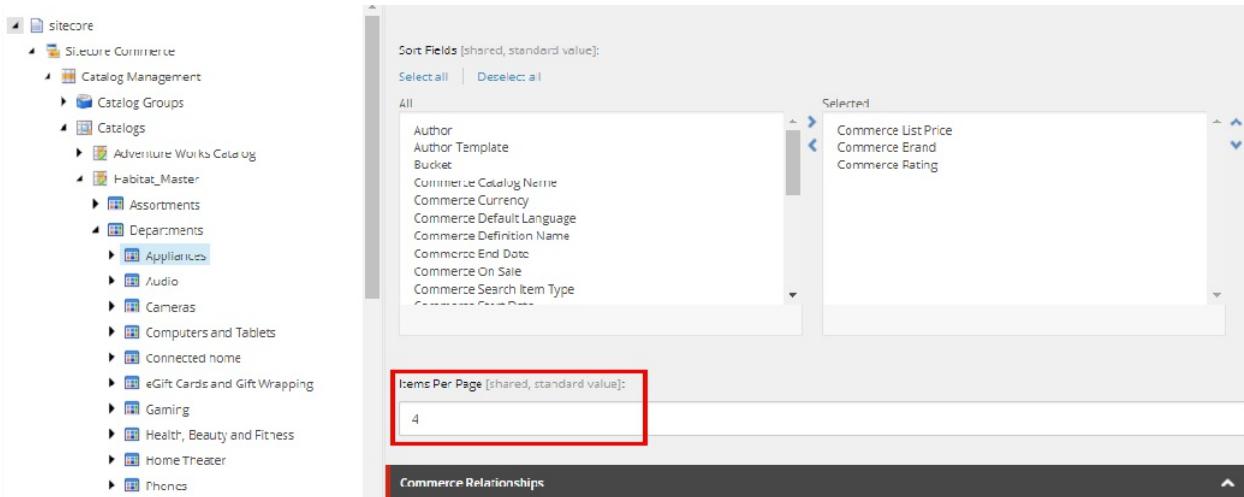
In the Commerce Control Panel, you can define a default value for all pages. Enter the overall default value in the *Catalog Configuration* item, in the Default Items Per Page field:

A screenshot of the Sitecore Commerce Control Panel. On the left, the navigation tree shows "sitecore" > "Commerce" > "Commerce Control Panel" > "Catalog Configuration". The right pane shows the "Catalog Configuration" item under "Commerce Control Panel". It includes fields for "Catalog [shared]:" (set to "Habitat\_Master"), "Use Index File For Product Status In Lists [shared]:" (unchecked), and "Default Items Per Page [shared, standard value]:" (set to "12", which is highlighted with a red box).

Commerce Control Panel	
Catalog [shared]:	Habitat_Master
Default Items Per Page [shared, standard value]:	12

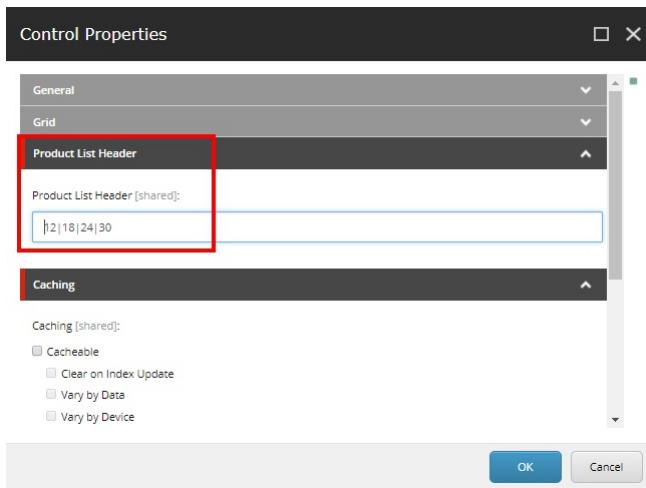
This is the number of items that will appear on the page until the user selects a different value.

You can configure the value in the Items Per Page field for a specific category page on the *Category* item. This value overrides the default from the catalog configuration. In the following example, the default number of items that will appear on a page for the Appliances category has been defined as 4:



## Options in the drop-down menu

You can change the options included in the Results per page drop-down menu by changing the settings in the Control Properties dialog for the rendering:



## Product List Page Info

If you place the Product List Page Info rendering above the Product List rendering on category and search pages, the rendering indicates the total number of items in the category or search, and it indicates which items from the list are shown on the current page.

Showing 13 - 20 of 20

## Product List Sorting

If you place the Product List Sorting rendering above the Product List

rendering on category and search pages, the customer can use the rendering to sort the category or product list.



## Sort options

The options available in the Sort by drop-down menu are the sort fields that have been defined for the category item. You can configure the sort fields on the *Category* item. The following is an example of the defined values for the Appliances category in the Habitat catalog:

The screenshot shows the Commerce Search interface with two panels. The left panel displays a list of sort fields under 'Sort Fields [shared, standard value]'. The right panel shows the same list with 'Commerce Brand' highlighted in the 'Selected' column. A red box highlights the 'Selected' column in the right panel.

All	Selected
Author	Commerce Brand
AuthorTemplate	
Bucket	
CommerceListPrice	
CommerceSearchItemType	
CommerceSelectableItemColor	
CommerceSelectableItemSize	
CommerceTags	
CommerceContent	
CreationDate&Author	
DateRange	

The Sort by field lists the sort criteria in the Selected field, with an ascending (Asc) option and a descending (Desc) option for each.

This gives the following result on the live storefront:

The screenshot shows a product grid on a storefront. The first product is a 'Habitat Dwell 4-Slice Toaster...' (Dwell Lifestyle Appliances). The second is a 'Habitat Dwell Pizzamaker Oven' (Dwell Lifestyle Appliances). The third is a 'Habitat Dwell 6-Cup Food Pr...' (Dwell Lifestyle Appliances). Each product has a '59.00 USD' price and a 'Details' button.

## Product List Pager

You can use the Product List Pager rendering on category and search pages. It allows the customer to navigate through the product list. The current page is highlighted in the Product List Pager rendering. The customer can click a page number or they can click Next » or Previous « to navigate between pages.



In the SXA Storefront, this rendering appears twice on each category and search page – above and below the Product List rendering.

## Product Price

You can use the Product Price rendering to display the product price. If the product is on sale, the discounted price is displayed, and the original price is displayed in strikethrough font.

12.95 USD ~~14.95 USD~~

## Product Variants

The Product Variants rendering displays a drop-down menu for each variant field (for example, one for size and one for color). If the customer selects an invalid combination, an error message is displayed in the message area and the Add to cart button is disabled.

In the Experience Editor, the rendering displays only a placeholder.

[Product Variant]

### Note

In the case of gift cards, the value of the card can be a variant.

## Promoted Products

The Promoted Products rendering displays a list of products in a format similar to the [Product List](#) rendering.

There are three types of lists that can be displayed in the Promoted Products rendering:

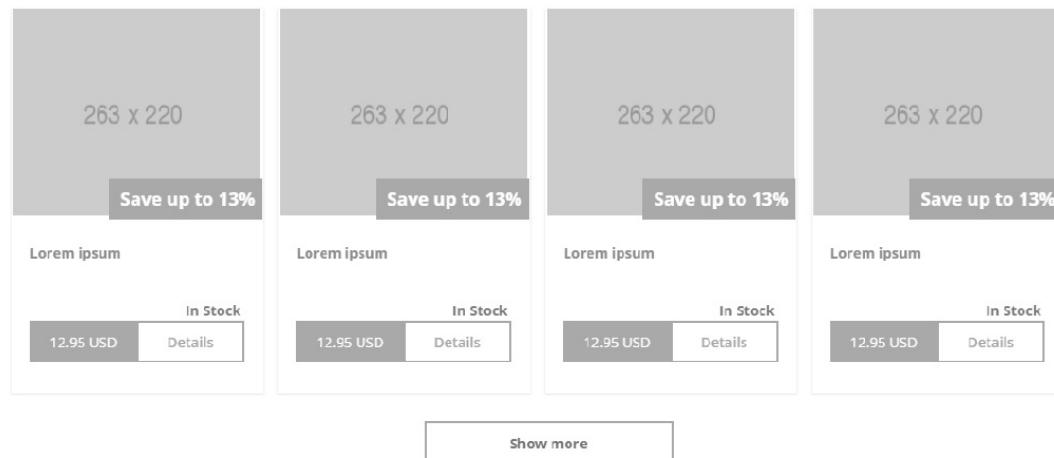
- Named search – a dynamic list of products that is created by searching the product catalog at runtime.

- Selected products – a static list of selected products.
- Related products – a list based on relationship data from your commerce system.

In the SXA Storefront, the home page contains an example of a promoted products list that was configured for the page, and the product details page contains an example of a related products list based on relationships in the catalog.

When you insert the Promoted Products rendering in the Experience Editor, it contains placeholder data:

Lorem ipsum dolor sit amet



### *Configuration*

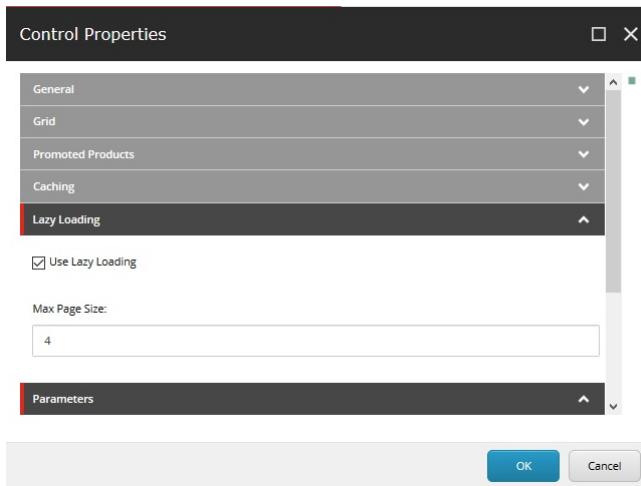
After you add the Promoted Products rendering to a page, you must configure it. You [create a Product List and configure the Promoted Products](#) rendering to display it.

### *Lazy loading*

You enable lazy loading to make a Show More button appear underneath a list of products, as illustrated in the previous screenshot of the Promoted Products rendering. When a visitor to your storefront clicks Show More, more items are loaded and displayed in the list. This is an alternative to displaying a fixed number of items at a time.

You can configure lazy loading for either the Promoted Products rendering or the Product List rendering. In the Control Properties dialog box for the rendering, under Lazy Loading, select the Use Lazy Loading check box. In

the Max Page Size field, indicate how many items you want to appear in the list when the visitor arrives on the page. This is also the number of items that are added to the list each time the visitor clicks Show More.



## Note

Normally, you *either* enable the Show More button *or* insert a product list header that includes the [Product List Items per Page](#) rendering. They are not designed to be used together, and the lazy loading settings override the selection in the Results per page field if they conflict.

## Search Initiated, Visited Category Page, and Visited Product Page

You can place the Search Initiated, Visited Category Page, and Visited Product Page renderings on pages to track visitor behavior. Each of these renderings makes an individual call to the Commerce Connect API to trigger an associated page event in xDB. You can view the resulting data in both Experience Profile and Experience Analytics.

These renderings are invisible on the live storefront, but they appear on the page in the Experience Editor so that they can be located and edited. For example, the Search Initiated rendering appears like this in the Experience Editor:

Search Initiated Page Event - Invisible at runtime.

When a search is initiated from the search bar on a page with the Search Initiated rendering, the rendering triggers the Search Initiated page event in Commerce Connect. In the SXA Storefront template, the Search Initiated

rendering is included on the Default Commerce Header partial design.

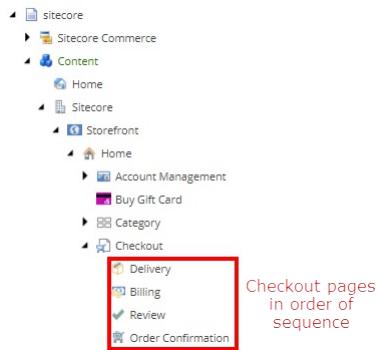
When a customer visits a page with the Visited Category Page rendering, the rendering triggers the CategoryPageVisited page event in Commerce Connect. In the SXA Storefront template, the Visited Category Page rendering is included on the category pages, in the main content area.

When a customer visits a page with the Visited Product Page rendering, the rendering triggers the ProductPageVisited page event in Commerce Connect. In the SXA Storefront template, the Visited Product Page rendering is included on the product detail pages, in the main content area.

## Commerce Checkout renderings

The Commerce Checkout group in the SXA toolbox contains a number of [renderings that you can use to create pages](#) to manage the checkout process.

The order of the pages in the Content Editor dictates the checkout page sequence. It determines which page is displayed as the first step of the checkout process, and which page is displayed when the customer clicks a button to move forwards or backwards in the process (for example, the Back to Delivery or Continue to Confirmation button).



This topic describes the following renderings in the Commerce Checkout group in the SXA Toolbox:

- Start Checkout
- Step Indicator
- Delivery
- Billing
- Review
- Order Confirmation

### Start Checkout

When the customer initiates the checkout process, they are directed to the *Checkout* page. This page must only contain the Start Checkout rendering, which does not have a visible output on the live site.

In the Experience Editor, the Start Checkout rendering looks similar to this:

[Start Checkout] On the runtime site, you will be redirected to the delivery page.

The Checkout Step provider determines which page to redirect the customer

to. In the associated data item, you can configure the message that is displayed in the rendering.

#### *Behavior on the live storefront*

On the live storefront, the Start Checkout rendering redirects the customer to the first visible step of the checkout process.

#### Note

If the cart is empty, the customer is redirected to the *Shopping Cart* page.

#### Step Indicator

The Step Indicator rendering is used to indicate progress through the steps of the checkout process.



The following steps are defined by default:

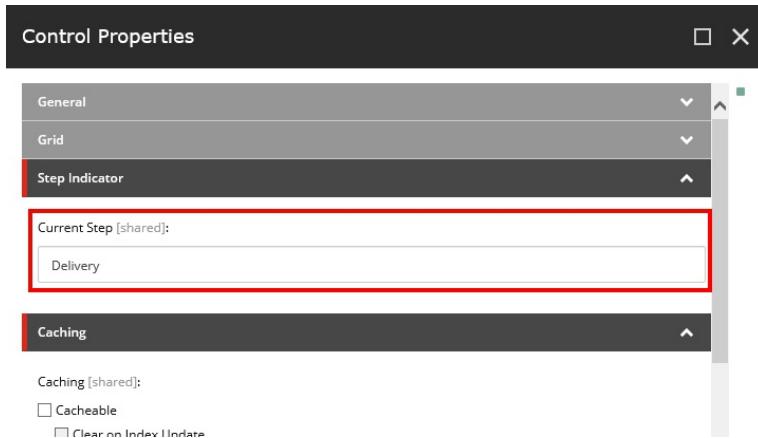
- Cart
- Delivery
- Billing
- Review

The order of the steps in the Step Indicator is determined by the configuration of the Step Name and Labels fields on the associated data item:

Step Indicator	
Step Name and Labels [unversioned]:	
Cart	Cart
Delivery	Delivery
Billing	Billing
Review	Review

The Step Name in the left column is the HTML Identifier of the step and the right column contains the text label that appears in the user interface.

As the customer moves through the checkout process, the current step is highlighted. On each checkout page, the current step is set in the Control Properties dialog box for the Step Indicator rendering:



In the Current Step field, enter the name of the step as it appears in the associated data item.

## Delivery

By default, delivery is the first visible step in the checkout process. The Delivery rendering lets the customer specify where and how items will be shipped. When you add the rendering to a page in the Experience Editor, it looks similar to this:



On the live storefront, after the customer selects a shipping method, enters an address, and clicks View shipping options, the Delivery rendering looks like this:

The screenshot shows the 'Delivery' step of a checkout process. At the top, there's a navigation bar with four tabs: 'Cart', 'Delivery' (which is highlighted in blue), 'Billing', and 'Review'. Below the tabs, there's a section titled 'Delivery preferences' with a dropdown menu set to 'Ship items'. The main area is labeled 'Shipping address:' and contains several input fields: 'Shipping name\*' (with 'John Smith' entered), 'City\*' (with 'Vancouver' entered), 'Country/Region\*' (with 'Canada' selected from a dropdown), 'State/Province' (with 'Alberta' entered), 'Address' (with '101 Main St.' entered), and 'Zip code' (with '65432' entered). Below this, a section titled 'Available shipping options:' lists four options: 'Ground', 'Next Day Air', 'Standard', and 'Standard Overnight'. The 'Standard' option is selected. At the bottom left is a button 'Continue shopping great products', and at the bottom right is a button 'Continue to billing'.

The options that the customer can select depend on the content of the cart. The following delivery options are available in the SXA Storefront solution, and you can customize them:

- Ship items – all items will be delivered to a single shipping address specified by the customer. When a valid address is entered, the following shipping options are available:
  - Ground
  - Next Day Air
  - Standard
  - Standard Overnight
- Ship items individually – lets the customer specify a different shipping address and shipping option for each item in the cart.
- Digital – if the customer orders a digital item, the customer must enter an email address for delivery.

When all items have a shipping address and delivery option, the Continue to billing button is enabled, and the customer can move to the next step of the checkout process.

In the Experience Editor, fields and buttons in the rendering are disabled.

On the live storefront, the customer can use the buttons to move backwards and forwards between steps in the checkout process. Delivery information that has been entered is saved when the customer clicks Continue to billing, and is retained if the customer then leaves the checkout process to continue shopping. By default, when the customer clicks the Continue Shopping button, the Home page is displayed. This is determined by the Checkout Step provider.

## Billing

The Billing rendering lets the customer specify payment information. The customer selects a billing option and enters billing information. Relevant fields are displayed, depending on the customer's selections.

Contact information

Email address *	Confirm email address *
<input type="text" value="Email address"/>	<input type="text" value="Confirm email address"/>
This field is required.	

▼Apply credit card

▼Apply gift card

Payment information	Order information
Payment total: 0.00 USD	Subtotal: 0
	Shipping cost: 0
	VAT: 0
	Savings: 0
	Order total: 0

[Back to delivery](#) [Continue to confirmation](#)

The rendering currently supports the following payment options:

- Federated payment via Braintree third-party payment control service
- Gift card payment

The customer can apply a gift card amount to an order, and they can pay the current balance by credit card.

For the billing address, there are the following options:

- Same as shipping address – only available if the Single shipment for all items option is selected on the delivery page. If multiple shipping

addresses have been specified, the customer must select from the registered addresses or enter a new one

- Select from registered addresses – all billing and shipping addresses that have been previously used for the account are displayed.
- Use another address – enter a new address.

The rendering uses the Braintree federated payment control to accept credit card payment. The customer enters the credit card information and then clicks Validate Payment to validate the amount.

To add a gift card payment, the customer expands the Apply gift card section, enters the gift card number, and presses Get Balance. If the gift card is valid, the balance is displayed. The customer can then apply the full amount of the order as a payment, or enter a smaller amount to be applied.

#### Note

If you intend to apply a gift card for part of the payment and then pay the balance using your credit card, you must enter the gift card first. The current remaining amount is applied to the credit card when you enter the credit card information, and you cannot change the credit card amount afterwards.

In the Experience Editor, fields and buttons in the rendering are disabled.

On the live storefront, once the payment information has been entered, the Continue to confirmation button is enabled, and the customer can move on to the next step of the checkout process.

The customer can use the buttons to move backwards and forwards between steps in the checkout process. Billing information that has been entered is remembered by Braintree. If the customer returns to checkout and the shopping cart and delivery information have not changed, the card information entered earlier can be used. However, if the shopping cart or delivery information have changed, the customer must reenter the card information.

#### Review

After the billing and delivery steps have been completed, the customer can review and confirm the order. The Review rendering summarizes the order, enabling the customer to review the current order and confirm the purchase.

In the Experience Editor, the rendering looks like this:

Order information

**Subtotal:** \_\_\_\_\_

Shipping cost: \_\_\_\_\_

VAT: \_\_\_\_\_

Savings: \_\_\_\_\_

Total : \_\_\_\_\_

Payment methods

Edit payment methods

[Back to billing](#)

[Confirm order](#)

On the live storefront, the rendering looks like this:

Order information

Subtotal:	192.84 USD
Shipping cost:	15.00 USD
VAT :	19.28 USD
Savings :	96.42 USD
Total :	130.70 USD

Shipping address

John Smith  
101 Main St.  
Vancouver, WA, US  
65432  
[Edit shipping address](#)

Billing address

John Smith  
101 Main St.  
Vancouver, WA, US  
65432  
[Edit billing address](#)

Payment methods

Credit card

-

130.70 USD

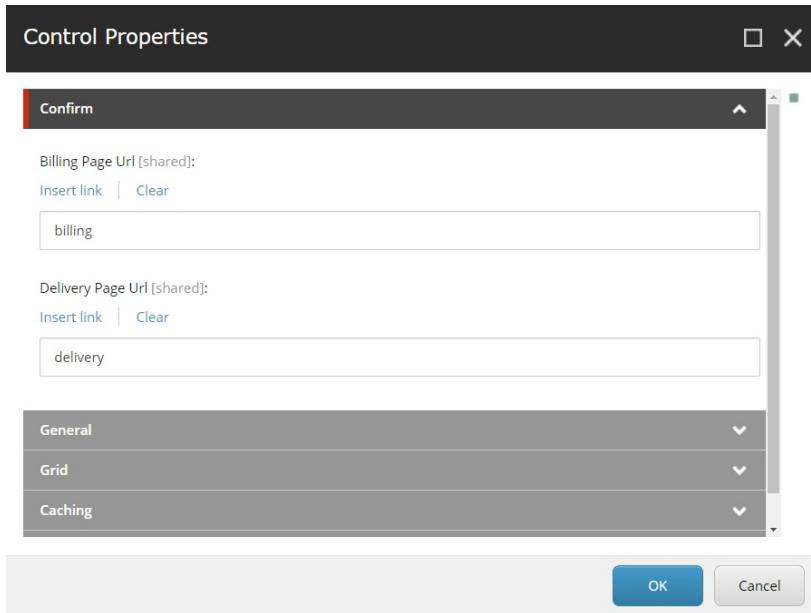
[Edit payment methods](#)

[Back to billing](#)

[Confirm order](#)

If the customer wants to place the order, the customer clicks Confirm order. By default, Confirm order brings the customer to the *Order Confirmation* page.

You use the Control Properties dialog box for the Review rendering to set the *Edit shipping address* and *Edit billing address* links.



## Note

If a visitor tries to bypass the client-side validation, for example, by directly accessing the URL for the page with the Confirm rendering and then submitting the order, then the Commerce engine will reject the order and an error will be displayed.

## Order Confirmation

The Order Confirmation rendering confirms that the customer order has been submitted.

In the Experience Editor, the rendering displays placeholder data for the order number and order status:

## Thanks for your order

Once your order has been shipped, we will notify you via email.

### Order information:

Your order confirmation number is: Lorem ipsum  
Order status: Lorem ipsum

Your receipt has been sent to the email you entered during checkout.

## Commerce Orders renderings

You can add the Commerce Orders renderings to your storefront to enable customers to see a list of their orders, and details for a single order. The order details and lists are only visible to customers who have logged into their account.

This topic describes the following renderings in the Commerce Orders group, in the SXA Toolbox:

- The Order History rendering and the Recent Orders rendering
- The Order Header, Order Lines, and Order Total renderings

### The Order History rendering and the Recent Orders rendering

The Order History rendering contains all of the orders placed by the customer, while the Recent Orders rendering displays only orders placed by the customer within the last 30 days. When you add either rendering to a page in the Experience Editor, it looks like this:

Order number	Status	Date	
UV3QSJDFL66P	lorem ipsum	01/01/1970	
VBB4SJHJM06T	lorem ipsum	01/01/1970	
CD7FJXXE02E	lorem ipsum	01/01/1970	
TMO45P6XXDR	lorem ipsum	01/01/1970	
FTP76DVP0X1	lorem ipsum	01/01/1970	

Both lists contain the following information for each order:

- Order Number – the order number assigned to the order by Sitecore Commerce.
- Status – the order's current status in the commerce system.
- Date –the date on which the order was placed.
- Order details icon – a link to a page with order details.

In the Experience Editor, the list contains placeholder data, and links are disabled.

### Configuration

For both renderings, you can configure text and tooltips on the data source item as usual, and you can define the destination of the order details link in the Control Properties dialog for the rendering.

In addition, on the *Recent Orders* data item, there are the following configuration options:

- You can display or hide the Recent Orders header.
- You can display or hide the View all orders link.
- You can determine the maximum number of orders displayed.

## The Order Header, Order Lines and Order Total renderings

In the Experience Editor, the Order Details page in the SXA Storefront template looks like this:

The screenshot shows the Order Details page with the following sections:

- Order Details**: A red box containing the **Order Header**.
- Order Header**: A red box containing the order number (xxxxx), date (1/1/0001), and status (Lorem ipsum).
- Product details**: A red box containing five order lines, each with a placeholder image (300x200) and placeholder product information (Lorem Ipsum). The columns are Unit price, Quantity, and Total.
- Order Lines**: A red box containing the five order lines with the following data:

Unit price	Quantity	Total
0.00 USD	1	0.00 USD Discount 0.01 USD)
0.00 USD	1	0.00 USD Discount 0.01 USD)
0.00 USD	1	0.00 USD Discount 0.01 USD)
0.00 USD	1	0.00 USD Discount 0.01 USD)
0.00 USD	1	0.00 USD Discount 0.01 USD)
- Order Totals**: A red box containing the order totals:

Subtotal	0.00 USD
Savings	0.00 USD
Shipping total	0.00 USD
VAT	0.00 USD
<b>Total</b>	<b>0.00 USD</b>
- View all orders**: A button at the bottom left.
- Print order**: A button at the bottom right.

### Order Header

The Order Header rendering contains general information about the order: the order number, date, and status. It contains placeholder data in the Experience Editor.

### *Order Lines*

You use the Order Lines rendering to display one line per product in an order, with the following details on each line: a product image, title, unit price, discount type, quantity, and total price. In the Experience Editor, the rendering displays placeholder data.

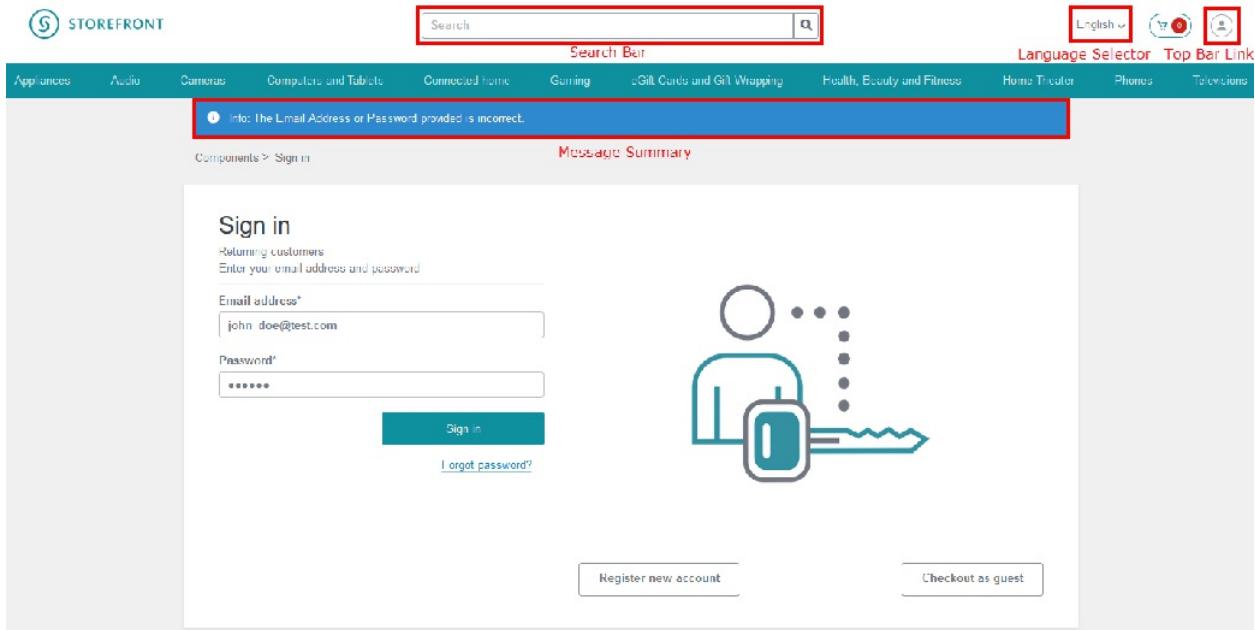
### *Order Totals*

You use the Order Totals rendering to display total amounts for an order. In the Experience Editor, no products are included in the order so the rendering displays 0 for all amounts.

## Commerce Shared renderings

You can use the renderings in the Commerce Shared group in the SXA toolbox for building different types of pages.

The following image is an example of how the Commerce Shared renderings might appear on a storefront page:



This topic describes the following renderings in the Commerce Shared group in the SXA toolbox:

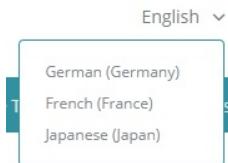
- Language Selector
- Top Bar Links
- Search Bar
- Message Summary

### Language Selector

When you add the Language Selector rendering to a page, it appears as a hyperlinked text that displays the current language. This rendering is similar to the Language Selector rendering in the Context group in the SXA toolbox. The difference is that you use the Language Selector rendering from the Commerce Shared group when you want to link the rendering to a language set in the Commerce Control Panel.

*Behavior on the live storefront*

On the live storefront, when you click the language selector link, the list of available languages expands. You can then select a language.



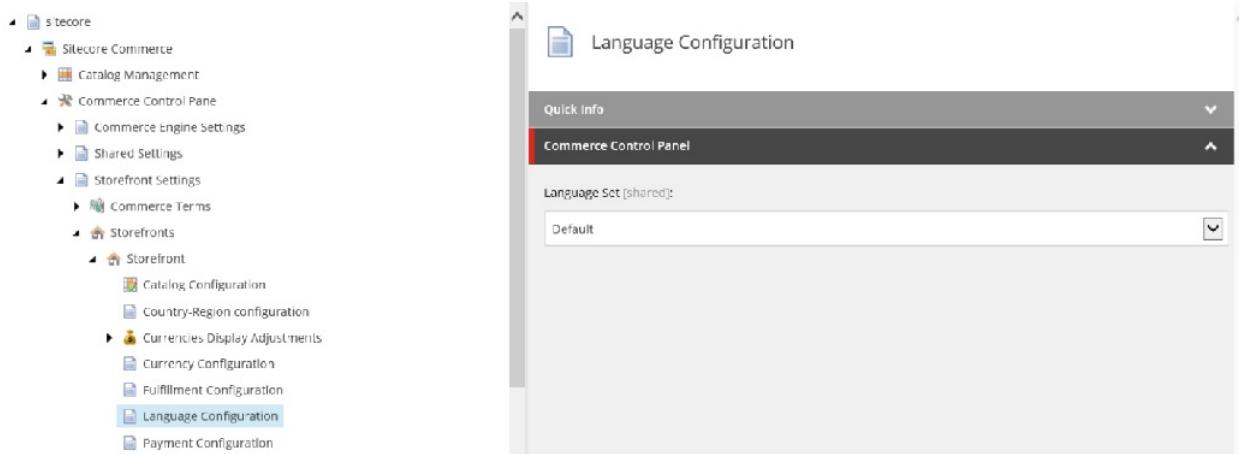
### Configuration

You can define the languages displayed in the Language Selector rendering in the [Language Set item](#) (`/sitecore/Commerce/Commerce Control Panel/Shared Settings/Language Sets`):

Each language set has a list of selected languages, and a default language:

A screenshot of the Sitecore Commerce Control Panel. At the top left, it says "Languages [shared]". Below that are two columns: "All" and "Selected". The "All" column is empty. The "Selected" column contains four language codes: "en", "de-DE", "fr-FR", and "ja-JP". At the bottom, there is a section labeled "Default Language [shared]" with a dropdown menu containing the value "en".

You specify the language set that applies to your storefront in: `sitecore/Commerce/Commerce Control Panel/Storefront Settings/Storefronts/<Your Storefront>/Language Configuration`, in the Language Set field:



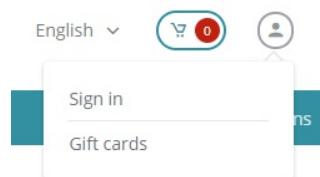
## Top Bar Links

The Top Bar Links rendering displays a number of links for the customer. The links displayed depend on the customer's authentication status. For example, if the customer is not logged in, a Sign in link is displayed; if the customer is logged in, the Sign out link is displayed together with the customer's login name.

### *Anonymous customer*

If the customer is not logged into their account, the Top Bar Links rendering displays the following links by default:

- Sign in
- Gift cards



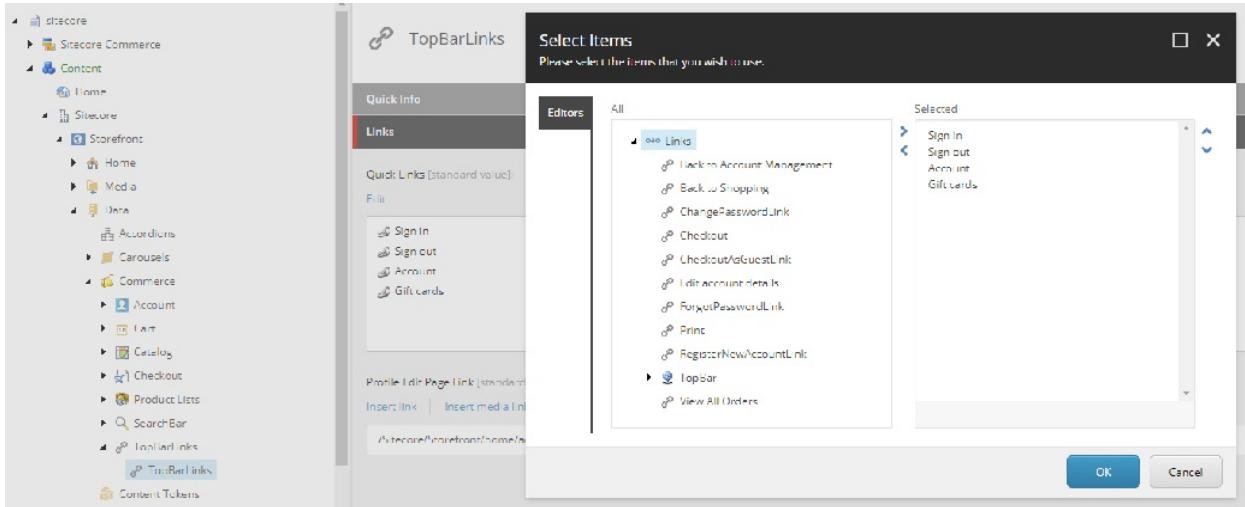
### *Logged in customer*

If the customer is logged in, the Top Bar Links rendering displays the following links by default:

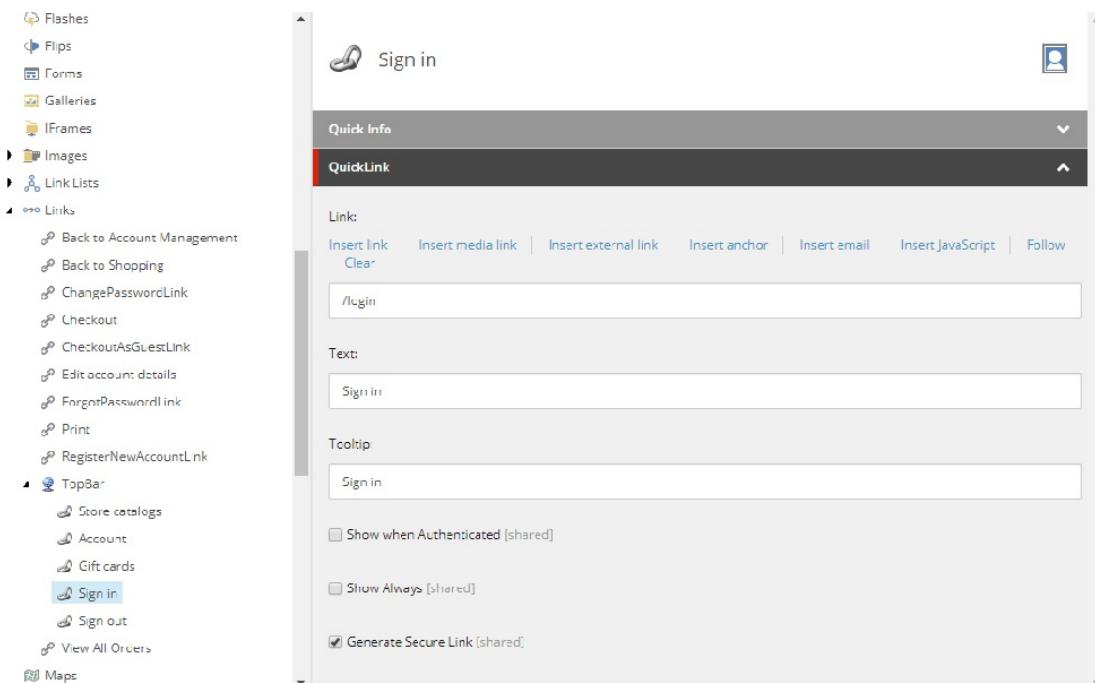
- Username
- Sign out
- Account
- Gift cards

## Configuration

In the associated content item, you can add or remove links from the list of links that are displayed in the rendering. Click the Edit link on the Links tab to open the Select Items dialog box:



The content items for the individual links can be found in:  
`/sitecore/content/Sitecore/[SITE]/Data/Links/TopBar`



For each link, you can configure the following fields:

- Link – leave the field blank if you want to display a static text.

- Text – enter the text of the link.
- Tooltip – enter a tooltip for the link.
- Show when Authenticated – select to only display the link when the customer is logged in.
- Show Always – select to always show the link.
- Generate Secure Link – select to enforce https:// when this link is clicked.

## Search Bar

When you place the Search Bar rendering on a page, it displays a text box with placeholder text and a search button identified by a looking-glass icon.

When the customer clicks the search button, the system redirects them to the search result page. The term that the customer entered in the text box is preloaded.

## Message Summary

The Message Summary rendering displays messages that have been triggered by other renderings and pushed to the Message Summary. The rendering groups messages by type and displays them in the following order:

- Errors
- Warnings
- Informational messages

The storefront style sheet (CSS) determines the appearance (style) of the different types of messages.

When you drop the Message Summary rendering on a page, it contains placeholder data:



On the live storefront, the rendering is invisible if it does not contain any messages.

## Configuration

- The messages displayed in the Message Summary rendering are

Commerce storefront system messages. You can configure the messages in the *System Messages* folder (`/sitecore/Commerce/Commerce Control Panel/Storefront Settings/Commerce Terms/System Messages`)

## Commerce meta renderings

The SXA storefront pages contain [meta renderings](#), which add metadata to the pages. The renderings are included in the meta partial designs that are used on every page in the SXA Storefront template.

Meta renderings are available in the SXA toolbox when you edit a meta partial design. In the Experience Editor, the Commerce Metadata partial design looks like the following:



```
<html>
<head>
    <title>[!title field is empty or is missing on the context item]</title>
    <meta content="Sample Meta Content" name="Sample Meta Name">
    <link href="" rel="shortcut icon">
    <meta content="Commerce Metadata for Catalog Items" property="og:title">
    <meta content="https://localhost:44301/Presentation/Partial-Designs/Commerce-Metadata-For-Catalog-Items.aspx" property="og:url">
    <meta content="["[metadescription field is empty or is missing on the context item]" name="description">
    <meta content="["[metakeywords field is empty or is missing on the context item]" name="keywords">
    <meta content="Commerce Metadata for Catalog Items" property="twitter:title">
    <meta content="summary_large_image" property="twitter:card">
    <meta name="viewport" content="width=device-width, initial-scale=1" /> Metadata rendering
    <link rel="canonical" href="https://sxa.storefront.com/category/sample_category">
    <title>Sample Title</title>
</head>
<body>
    <form id="__RequestVerificationToken" action="#" method="post"> <input name="__RequestVerificationToken" type="hidden" value="d74fkY21sNl0pe7x1H0IVYd-hD0xwh10h1...">
    <input name="_SiteVirtualFolder" type="hidden" value="/" /><input name="_SiteRootPath" type="hidden" value="/sitecore/content/Sitecore/Storefront"><input name="...>
</div>
<div id="header" class="main clearfix">
</div>
<div id="content" class="main clearfix">
...
</div>
<div id="footer" class="main clearfix">
...
</div>
<div class="cookie-notification permissive" style="display:none">
    <div class="close"> <a onclick="">X</a> </div>
    <div class="info"> Cookies help us improve your website experience. <br>By using our website, you agree to our use of cookies. </div>
    <div class="submit"> <a onclick="">Confirm</a> </div>
</div>
<div class="stickynotes" style="visibility: hidden"> </div>
</body>
</html>
```

This topic describes:

- The Antiforgery meta rendering
- The Metadata meta rendering
- The Site Bindings meta rendering

### The Antiforgery meta rendering

You can use the Antiforgery meta rendering to insert the `<form>` tag with the antiforgery token into a partial design. The antiforgery token is passed to every JavaScript API call that the Commerce renderings make. This prevents cross-site request attacks from being made on the website.

#### Note

The Commerce renderings are dependent on the Antiforgery meta rendering.

You must include the Antiforgery meta rendering on all storefront pages to ensure that the Commerce renderings behave as expected.

In the SXA Storefront template, the Antiforgery meta rendering is included in the <body> of the Commerce Metadata partial design so that it generates the antiforgery token HTML that is required by the site.

#### *Appearance on the live storefront*

The following shows the Antiforgery rendering in the HTML source for the home page:

```
56 <meta name="viewport" content="width=device-width, initial-scale=1"/>
57
58 </head>
59 <body class="home-page default-device">
60
61 <form id="_LHSForm" action="#" method="post">
62   <input name="__RequestVerificationToken" type="hidden" value="S0qTR_RgkbXfEaAEcDg1FxDFMX-cmKlopXOPErqOPEwvNkm4e1udWCoG3eqCQdIGKnVvXssQpuhDg4f68ZhurTuQ\3wleGgC5B54aBcC01" />
63 </form>
64
65 <!-- m-wrapper -->
```

#### The Metadata meta rendering

You can use the Metadata meta rendering to inject canonical URLs in order to improve the search engine indexing of categories and products.

#### *Appearance on the live storefront*

The following shows the Metadata meta rendering in the HTML source for the audio category page:

```
41
42 <meta content="" property="twitter:title" /><meta content="summary_large_image" property="twitter:card" />
43 <meta name="viewport" content="width=device-width, initial-scale=1"/>
44
45
46
47
48
49
50
51 <link rel='canonical' href='https://sxa.storefront.com/category/Habitat_Master-Audio' />
52 <title>Audio</title>
53
54
55
56
57
58 <link href="" rel="shortcut icon" />
59
60
61 <meta content="" property="og:title" /><meta content="https://test763ca2:443/Shop/-w-.aspx" property="og:url" />
62
63
64
65
```

#### The Site Bindings meta rendering

SXA provides [site management](#) features that enable you to manage site binding attributes and other related configuration.

After the Site Bindings rendering is added to a page, the metadata information about the site's current bindings is rendered in the HTML output as hidden fields. Client-side scripts can access those hidden fields. The Site

Bindings rendering inserts the following fields in the HTML source:

- SiteVirtualFolder – the current site's virtual folder.
- SiteRootPath – the current site's root item path.
- SiteStartPath – the start item for the current site.
- SiteContentStartPath – the content start item for the current site.

#### *Apearance on the live storefront*

The following shows the Site Bindings meta rendering in the HTML source for a page in the SXA Storefront:

```
44| <form id="_CRSFform" action="#" method="post">
45|   <input name="__RequestVerificationToken" type="hidden" value="yJuWZlUcwLaJqyCR_LyJcDKaUqzgq9_nGNelz/
46| 
47|
48|
49| <input name="_SiteVirtualFolder" type="hidden" value="/" />
50| <input name="_SiteRootPath" type="hidden" value="/sitecore/content/Sitecore/Storefront" />
51| <input name="_SiteStartPath" type="hidden" value="/sitecore/content/Sitecore/Storefront/home" />
52| <input name="_SiteContentStartPath" type="hidden" value="/sitecore/content/Sitecore/Storefront" />
53|
54|
55| <!-- #wrapper -->
56| <div id="wrapper">
```

## Using Commerce renderings for SXA

### Using rendering variants to display product information

The Commerce renderings for SXA include a Product Information rendering that is used to show the product title and description on the product details page. An alternative method to display this information is to use [variants of standard SXA renderings](#).

This topic describes:

- Rendering variants installed with the SXA Storefront
- The data source for the renderings
- The Page Content rendering
- Using variants of the Title rendering and Rich Text rendering to display product information

#### Rendering variants installed with the SXA Storefront

When you install the SXA Storefront, a few variants of SXA renderings are installed in the

/sitecore/content/Sitecore/Storefront/Presentation/Renderings/Variants folder. The following SXA renderings in the Page Content group have variants that you can use to display product information:

- Page Content—use to show title, description and/or other product fields in a single SXA component.
- Rich Text—use to display rich text or HTML fields, for example, product description.
- Title –use to display single-line text fields, for example, title or display name.

#### The data source for the renderings

The storefront uses a single page to display different products. The CatalogItemResolver pipeline processor sets the current product context.

To enable you to use product item fields in SXA rendering variants, the ResolveCatalogContext pipeline processor dynamically sets the data source for these renderings to the current product, based on the following

conditions:

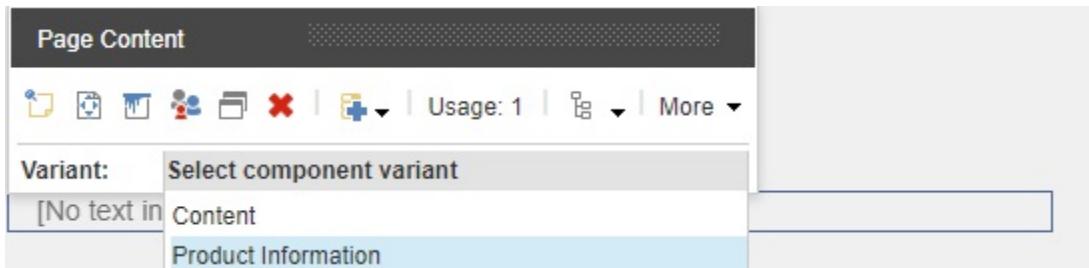
- The rendering is one of the following: Page Content, Rich Text, or Title.
- The product context is set.
- The data source for the rendering is not already set in the control properties.

## The Page Content rendering

The product page in the SXA Storefront uses a variant of the Page Content rendering to display product information. The Product Information rendering contains these fields:

- Display Name
- Item Number
- Description
- Features

You can insert the Page Content rendering on a page to display product information. You do this in the Experience Editor by dragging the rendering from the toolbox onto the page and then selecting the Product Information variant from the drop-down menu.



### *Behavior of the rendering variant in the Experience Editor*

In the Experience Editor, if a product has been selected, the Product information variant displays the product information from the commerce system.

The screenshot shows the 'Page Content' rendering variant in a content editor. At the top, there are standard toolbar icons and status information: 'Workflow: No', 'Usage: 1', and a 'More' dropdown. Below that, the 'Variant:' dropdown is set to 'Product Information'. The main content area displays the product title 'Habitat Dwell 2-Slice Toaster and Oven Combination'. Underneath the title, the text 'ItemNumber field' is displayed in red, indicating it is a placeholder. A larger block of placeholder text follows: 'Make your favorite toasted items and baked foods with this Helix Dwell Combination toaster and oven that features an extra-wide toast slot to accommodate larger breads and bagels.' At the bottom of this section, the text 'Features field' is also displayed in red. The entire content area is enclosed in a red border.

Unlike the Commerce Product Information rendering, the Page Content rendering variant cannot display placeholder data. If data is not available for a field, then the Page Content rendering displays the field name, for example, Features field. If no product is selected, the product title is displayed as \* and field names are displayed for all fields.

The screenshot shows a live storefront page for the 'Habitat Dwell 2-Slice Toaster and Oven Combination'. The product title is visible. Below it, a large red box highlights the placeholder text for three fields: 'ItemNumber field', 'description field', and 'Features field'. At the bottom of the page, there is a stock status message: 'In-Stock : 74 Items in stock' with a smiley face icon. The price is listed as '12.95 USD' crossed out with '14.95 USD' below it. A small note at the bottom left says '[Product Variant]'. The entire content area is enclosed in a red border.

#### *Behavior of the rendering variant on the live storefront*

On the live storefront, the product information is displayed. If there is no data available for a field, the field is hidden.

The screenshot shows a live storefront page for the 'Habitat Dwell 2-Slice Toaster and Oven Combination'. The product title is displayed. Below it is a summary paragraph: 'Make your favorite toasted items and baked foods with this Helix Dwell Combination toaster and oven that features an extra-wide toast slot to accommodate larger breads and bagels.' The entire content area is enclosed in a red border.

#### Using variants of the Title rendering and Rich Text rendering to display product information

Another way to display product information is to use two rendering variants.

- To display the product name, insert the Title rendering and select the Display Name Heading variant.

- To display the product description, insert the Rich Text rendering and select the Description variant.

## Configure a product list and display it on your storefront

You can configure product lists with promoted or related products. To display a list on a page, you insert the Promoted Products rendering on the page and then assign the list to the rendering. In the SXA Storefront, the Promoted Products rendering on the home page displays promoted products, and the Promoted Products rendering on the Product Details page displays related products.

There are three types of lists that can be displayed in the Promoted Products rendering:

- Named search – A dynamic list of products that is created by searching the product catalog at runtime.
- Selected products – A static list of selected products.
- Related products – A list based on relationship data from your commerce system.

This topic describes how to:

- Create a Named Search list
- Create a Selected Products list
- Add a product list to a page
- Display a list of related products

### Create a Named Search list

1. In the Content Editor, navigate to the `/sitecore/content/[your tenant]/[your site]/Data/Commerce/Product Lists` folder and insert an item based on the *Named Search* template in `/sitecore/templates/Foundation/Experience Accelerator/Commerce`.
2. Assign a name to the search and click OK.
3. In the Default bucket query field, [build a query](#).

Search Content

Mira Laptops

Quick Info

Commerce Search

Named Search

Advanced

Appearance

Help

Indexing

Insert Options

**Item Buckets**

Bucket parent reference - Reserved for system use. Do not set manually [shared]:

Is Bucket - Reserved for system use. Do not set manually [shared]

Bucketable - Can be stored as an unstructured term in an item bucket [shared]

Lock child relationships - Maintains the hierarchical relationship between this item and its children when this item is stored in an item bucket [shared]

**Default bucket query [shared]:**

Build query | Paste query | Clear

```
+commercesearchitemtype:product&brand:Mira Laptops&location:(C00FC07F-DAC9-4C11-90C5-5D19047399CF)&
```

#### 4. To test the query, click the Search tab.

Search Content

Commerce search itemtype:product & brand:Mira Laptops & location:(C00FC07F-DAC9-4C11-90C5-5D19047399CF) &

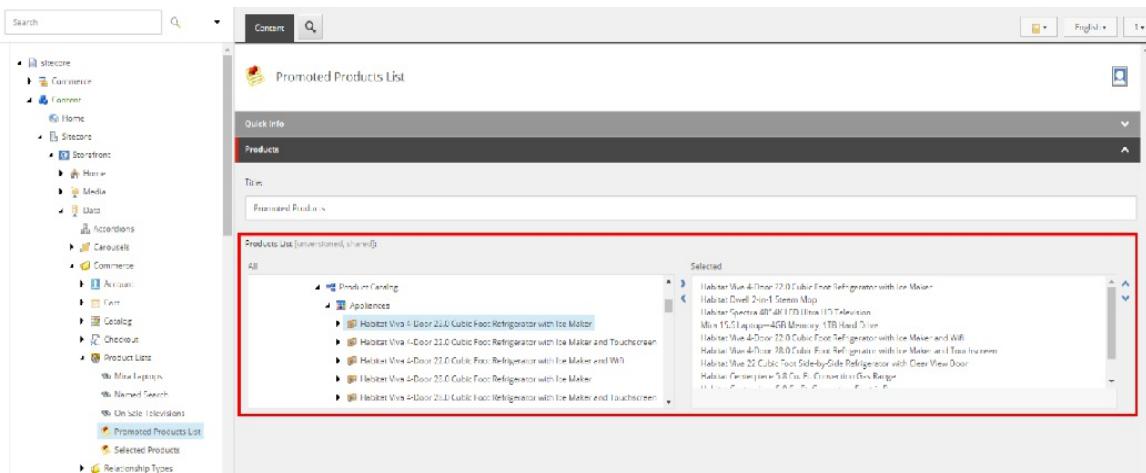
44 Results

Result	Template	Location	Version	Created	By	Language
6042175(4)	Template:Laptop	Computers and Tablets	Version:1	Created: 2016-12-26	By: sitecore\anonymous	de-DE
6042175(4)	Template:Laptop	Computers and Tablets	Version:1	Created: 2016-12-26	By: sitecore\anonymous	en
6042175(4)	Template:Laptop	Computers and Tablets	Version:1	Created: 2016-12-26	By: sitecore\anonymous	fr-FR
6042175(4)	Template:Laptop	Computers and Tablets	Version:1	Created: 2016-12-26	By: sitecore\anonymous	ja-JP
6042176(4)	Template:Laptop	Computers and Tablets	Version:1	Created: 2016-12-26	By: sitecore\anonymous	de-DE
6042176(4)	Template:Laptop	Computers and Tablets	Version:1	Created: 2016-12-26	By: sitecore\anonymous	en
6042176(4)	Template:Laptop	Computers and Tablets	Version:1	Created: 2016-12-26	By: sitecore\anonymous	fr-FR
6042176(4)	Template:Laptop	Computers and Tablets	Version:1	Created: 2016-12-26	By: sitecore\anonymous	ja-JP
6042177(4)	Template:Laptop	Computers and Tablets	Version:1	Created: 2016-12-26	By: sitecore\anonymous	de-DE

- Save the item.

## Create a Selected Products list

- In the Content Editor, navigate to the `/sitecore/content/[your tenant]/[your site]/Data/Commerce/Product Lists` folder and insert an item based on the *Selected Products* template in `/sitecore/templates/Foundation/Experience Accelerator/Commerce`.
- Assign a name to the list and click OK.
- In the Products List field, navigate to an item in the product catalog and double-click on the item to add it to the list. Repeat for all the products that you want to include in the list.



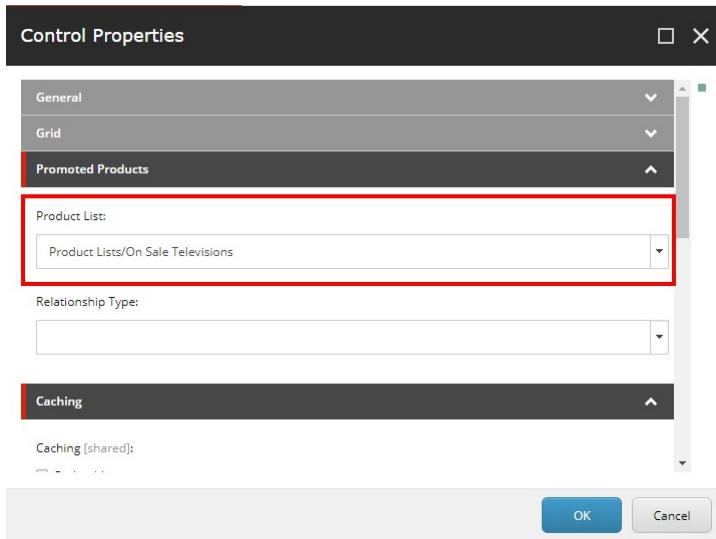
- Save the list.

## Add a product list to a page

After you configure a Named Search list or a Selected Products list, you can display it on a page.

To add a product list to a page:

- Insert the Promoted Products rendering on a page.
- In the Control Properties dialog box, select a product list in the Product List drop-down menu:



3. The following fields on the associated content item contain default text that you can change:

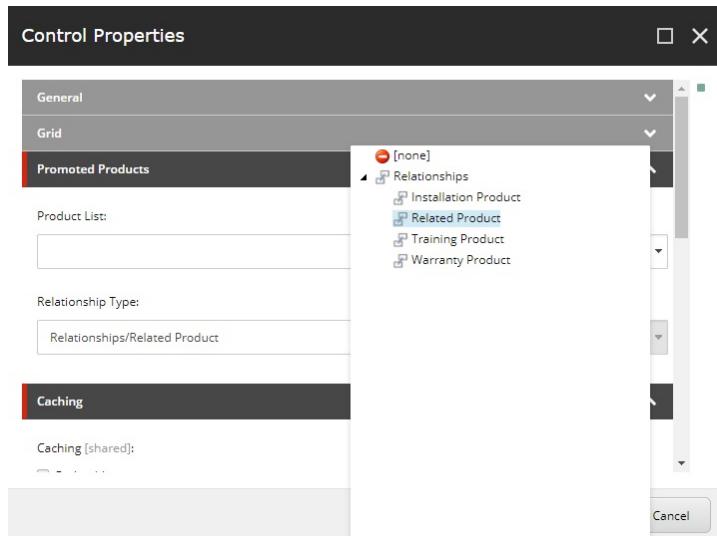
- Save Percent Lead – Specify the text that will be displayed when there is a product with a discount. This text appears before the amount of saving for each product.
- Product Page Link Text – Specify the display text for the link to the product details page. The user activates the link by clicking the product name or the link with this text.
- Category – Specify the text that will appear before a category of products.
- Price Starting From Text – Specify the text that will appear at the top of the Price component when there are product variants at different prices. This text indicates that the displayed price is the lowest variant price.

## Display a list of related products

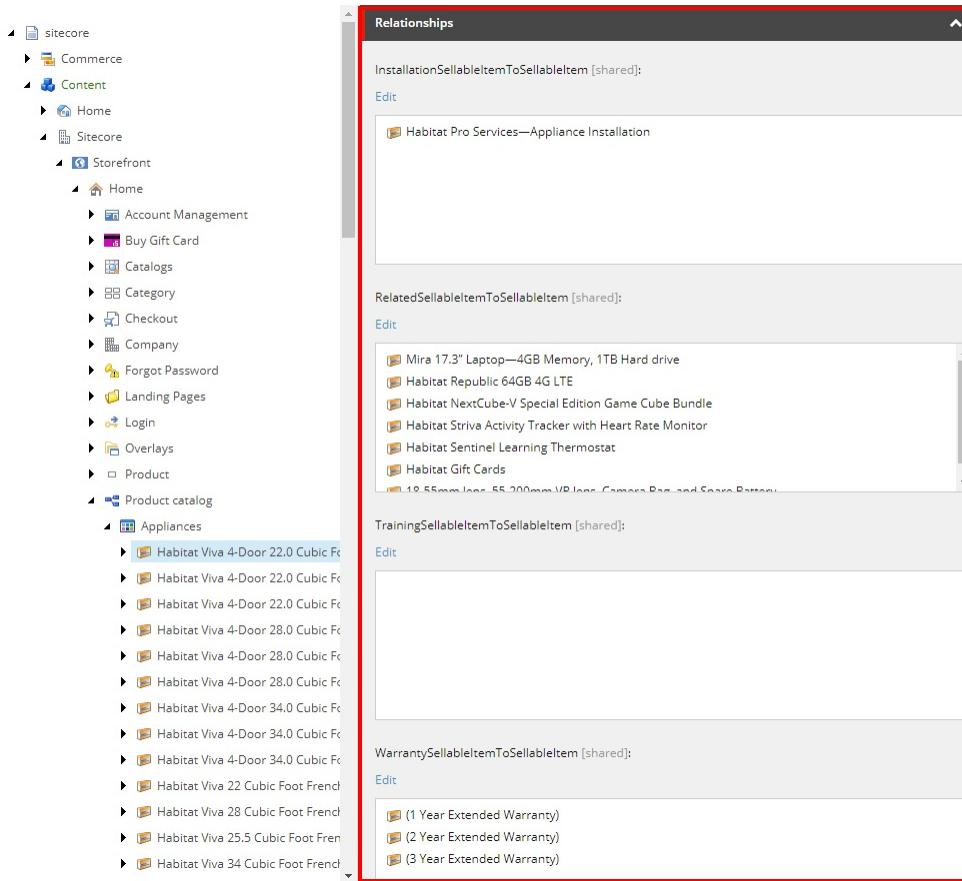
If your commerce system contains data about related products, you can add a list of related products to a product page. You can choose which type of relationship the products in the list have to the product on the page.

To display a list of related products:

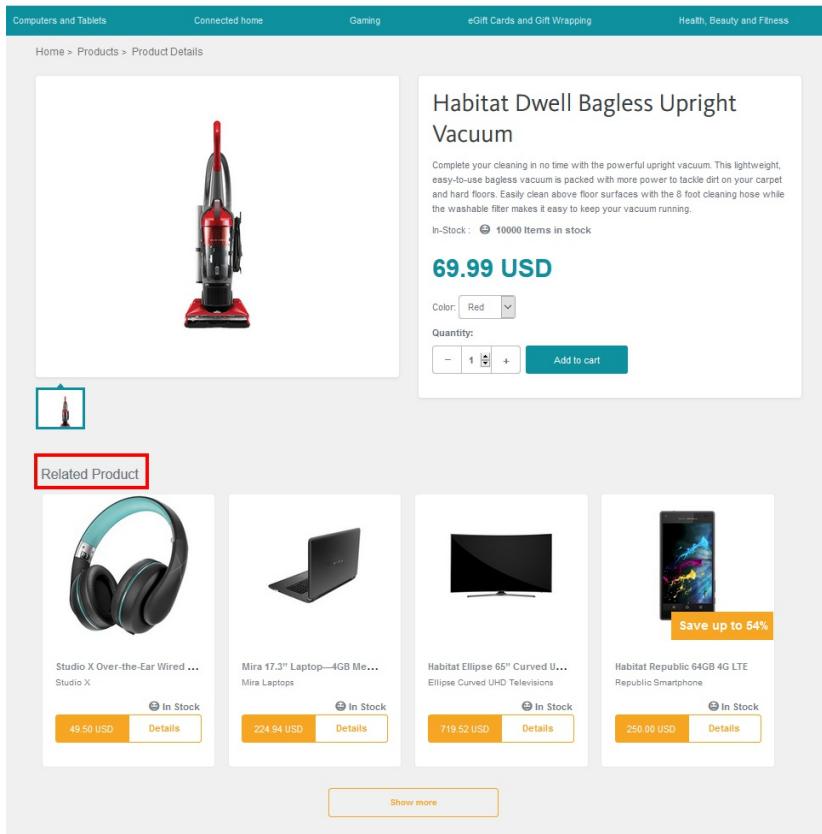
1. Insert the Promoted Products rendering on a page that has a current catalog item, for example, a product details page.
2. In the Control Properties dialog box for the rendering, select an option from the Relationship Type drop-down menu:



The rendering shows placeholder data in the Experience Editor, but on the live storefront it will display products that are related to the current product. The list of products is derived from the corresponding field on the Relationships tab on the catalog item.



On the live storefront, a header appears above the list:



From SXA Storefront 1.0.2, you can configure the header text for each relationship type on the respective relationship type item, in the `/sitecore/content/[your tenant]/[your site]/Data/Commerce/Relationships` folder:

- ▶ Product Lists
- ◀ Relationships
  - ▶ Installation Product
  - ▶ Related Product
  - ▶ Training Product
  - ▶ Warranty Product
- ▶ Search Bar
- ▶ Top Bar Links
- ▶ Content Tokens

**Quick Info**

**Relationship Type**

**Relation Title:**

**Field Name [shared]:**

### Note

Each instance of the Promoted Products rendering can only display one list. If you fill in both the Product List field and the Relationship Type field in the Control Properties dialog box for a single component, only the product list is displayed. To include more than one list on a page, you must insert multiple instances of the Promoted Products rendering on the page.



## Technical Overview

### The SXA Storefront installation packages

This topic describes the packages that you can use to install the SXA Storefront. When you install the SXA Storefront installation packages, you make it possible to select SXA Storefront features when you [create new tenants and sites](#) in SXA.

This topic describes the content of the following SXA Storefront installation packages:

- Sitecore Commerce Experience Accelerator
- Sitecore Commerce Experience Accelerator Storefront
- Sitecore Commerce Experience Accelerator Storefront Themes
- Sitecore Commerce Experience Accelerator Habitat catalog

#### Sitecore Commerce Experience Accelerator

This package contains:

- The foundation layer for all SXA Storefront features.
- Commerce renderings for the SXA Toolbox.
- The scaffolding functionality to create a blank tenant and site.
- The extension theme, which SXA uses to create new custom themes that support the commerce components.

#### Important

This package is required if you install any of the other SXA Storefront packages.

#### Sitecore Commerce Experience Accelerator Storefront

This package contains the storefront template, which consists of:

- Prebuilt pages.
- Scaffolding items that Sitecore uses to create a storefront based on the storefront template.

After installation, there is no content in the `/sitecore/content` folder. All scaffolding items are in branches, from where you can deploy them using scaffolding.

### [Sitecore Commerce Experience Accelerator Storefront Themes](#)

This package contains the following SXA Storefront themes: *Storefront Mock-up with product images*, *Storefront Mock-up with mock images*, and *Storefront Branded*.

### [Sitecore Commerce Experience Accelerator Habitat catalog](#)

This package contains the Habitat catalog and the corresponding scaffolding items.

## Enforcing HTTPS on Commerce-related pages

We have included a processor in the

`Sitecore.Commerce.XA.Foundation.common.config` configuration file. This processor is part of the `<httpRequestBegin>` pipeline, which enforces HTTPS on all Commerce-related pages when running a live storefront.

If a visitor tries to use HTTP to access a page that was created with the `_CommercePage` foundation template, the processor redirects the page to HTTPS.

The pipeline does not enforce HTTPS in the Experience Editor, or if the page was not created with the `_CommercePage` foundation template.

### Configuration

The processor is defined in the

`Sitecore.Commerce.XA.Foundation.common.config` configuration file as follows:

```
<httpRequestBegin>
    <processor
        type="Sitecore.Commerce.XA.Foundation.Common.Pipelines.SecurityProcessor,
        Sitecore.Commerce.XA.Foundation.Common"
        patch:after="processor[@type='Sitecore.Pipelines.HttpRequestBegin']" />
</httpRequestBegin>
```

If you do not want to enforce HTTPS on a specific storefront site, you can disable the Enforce SSL field in the [Commerce Control Panel settings for the storefront](#) (in `/sitecore/Commerce/Commerce Control Panel/Storefront Settings/Storefronts/<storefront name>/Storefront Configuration`).

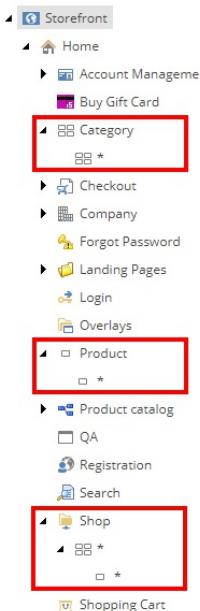
If you do not want to enforce HTTPS on *any* sites in your installation, you can remove the processor from the config file.

## Shop pages and canonical URLs

It is important to assign canonical URLs to pages on your storefront to improve how your storefront ranks on search engines.

The SXA Storefront sample site has the following dynamic pages:

- The canonical Category page (Category/\*)
- The canonical Product detail page (Product/\*)
- The Category page for site navigation (Shop/\*)
- The Product detail page for site navigation (Shop/\*/\*)



When a customer browses your storefront, they might see different URLs for the same page, depending how they navigate to the page. You must assign a canonical URL to each page. You can choose which terms to use for **product**, **category**, and **shop** in the URLs.

This topic describes:

- Shop page URLs and canonical URLs
- Renaming the Product, Category, or Shop page

### Shop page URLs and canonical URLs

The Shop/\* and Shop/\*/\* pages display the category and product pages within the navigation structure of your website. When a customer navigates

on your website, they see `shop` as part of the URL path, for example:  
[HYPERLINK https://<server>/shop/Audio%3dhabitat\\_master-audio](https://<server>/shop/Audio%3dhabitat_master-audio).

To improve the search engine ranking of your website, it is important you assign canonical URLs to `shop` pages.

#### Note

If you create your storefront site using the Commerce Storefront Template feature, then a canonical URL is automatically included in the metadata for every storefront page.

If you create your own page designs, you can insert the Commerce Metadata meta rendering from the SXA toolbox in your page designs in order to assign canonical URLs to every page.

The following tables show the navigation URL and the canonical URL for two pages in the SXA Storefront:

<b>Audio category page</b>	
URL after navigating from the home page	
Canonical URL	<a href="https://&lt;server&gt;/category/Habitat_Master-Audio">https://&lt;server&gt;/category/Habitat_Master-Audio</a>

<b>Bookshelf speakers product detail page</b>	
URL after navigating from the home page	
Canonical URL	<a href="https://&lt;server&gt;/product/6042080">https://&lt;server&gt;/product/6042080</a>

#### [Renaming the Product, Category, or Shop page](#)

As demonstrated in the previous section, the Product, Category and Shop page names appear in URLs for the SXA storefront. If you change the name of the Product, Category or Shop page in the Content Editor, you must update

the catalog configuration in the `/sitecore/Commerce/Commerce Control Panel/Storefront Settings/Storefronts/<Storefront>/Catalog Configuration` item, in the Shop Pages Root Paths section.

The screenshot shows the Sitecore Commerce Control Panel. On the left, there is a navigation tree under 'SITEMAP':

- Sitemaps
- Sitecore Commerce
  - Commerce Engine Settings
  - Shared Settings
  - Storefront Settings
    - Commerce Terms
    - Global Configuration
- Commerce Control Panel
  - Commerce Engine Default Storefront
  - Storefront
    - Access Management Configuration
    - Catalog Configuration (highlighted with a red box)
    - Country-Region Configuration
    - Currencies Display Adjustments
    - Currency Configuration
    - Fulfillment Configuration
    - Language Configuration

The main content area is titled 'Catalog Configuration'. It contains sections for 'Quick Info', 'Commerce Control Panel', and 'Gift Card'. The 'Shop Pages Root Paths' section is highlighted with a red box and contains the following fields:

- Shop Page Root Path - Relative path for the shop page root [shared]: `shop`
- Category Page Root Path - Relative path for the category page. This will be used for direction navigation and canonical URL generation [shared]: `category`
- Product Page Root Path - Relative path for the product detail page. This will be used for direction navigation and canonical URL generation [shared]: `product`

### Note

After you have updated the configuration and published the site, if you want to see the results in the storefront, you must clear your browser cache before navigating to the site.

## Caching SXA Storefront renderings

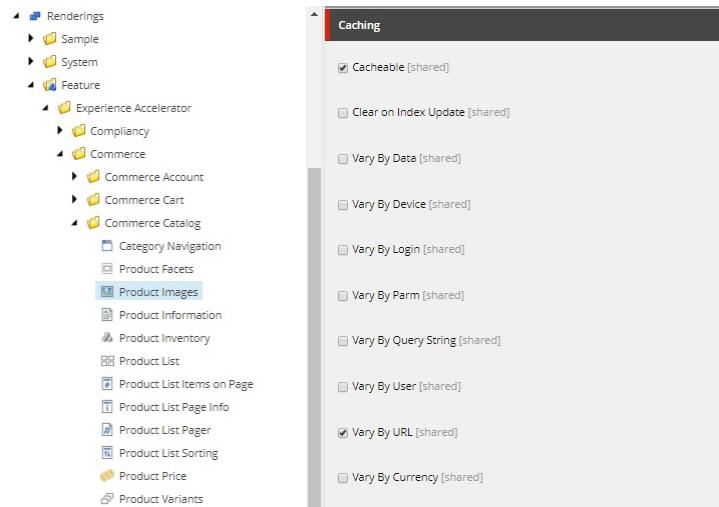
Sitecore supports caching and you can control when cached data is updated by changing the caching settings for renderings or components.

This topic describes:

- Caching settings on renderings
- Caching settings on components
- Default caching settings for Commerce renderings for SXA
- Default caching settings for SXA renderings in the SXA Storefront

### Caching settings on renderings

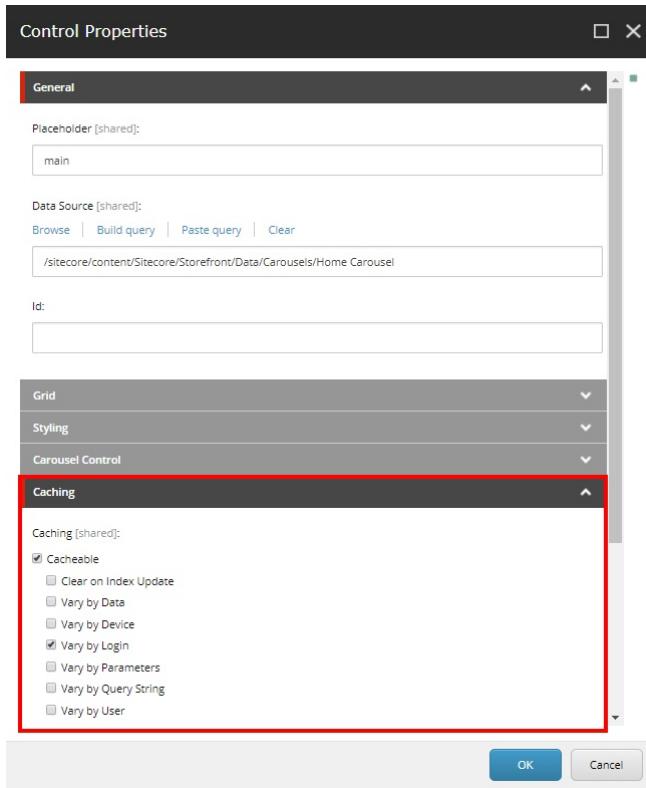
You can configure caching settings on the rendering item in the Content Editor. For example, you can find the Commerce renderings for SXA Storefront in the */sitecore/layout/Renderings/Feature/Experience Accelerator/Commerce* folder:



If you use a rendering on multiple pages or partial designs, these settings apply to all instances of the rendering unless the settings are overwritten for a specific component.

### Caching settings on components

You can configure caching settings for a component in the Control Properties dialog box for the component. You can do this in the Experience Editor:



These settings only apply to the component on one page or partial design – you configure the caching settings separately for each component that uses the rendering.

#### Note

The Vary by URL and Vary by Currency options do not appear in the Control Properties dialog box. To select these options, you must configure the settings on the rendering item.

#### Default caching settings for Commerce renderings for SXA

The default caching settings for Commerce renderings for SXA are set on the rendering items in the Content Editor.

The Commerce renderings in the following table are all configured as Cacheable by default. The following table indicates the other default caching settings for each rendering:

Rendering	Clear on Index Update	Vary By URL	Vary By Currency	Vary By Login	Vary By Query String

Category Navigation	x				
Subcategory Navigation		x			
Product Facets		x			
Product Images		x			
Product Information		x	x	x	
Product List	x	x	x		
Product Recommendation	x		x		
Related Items		x			
Language Selector					x
Search Bar					x

### Note

We recommend that you do not change the caching settings. The default settings take the context of each rendering into account. For example, each product-related rendering is used to display many different products. If the Vary By URL option is not selected, the cached output for the wrong product can be displayed. For example, in the following scenario:

The customer views product A

([https://sxa.storefront.com/shop/Appliances\\_appliances/HabitatDwell2-SliceToasterandOvenCombination\\_6042855](https://sxa.storefront.com/shop/Appliances_appliances/HabitatDwell2-SliceToasterandOvenCombination_6042855)) and

Sitecore caches the output. Then the customer navigates to product B

([https://sxa.storefront.com/shop/Appliances\\_appliances/shop/Appliances\\_appliances/HabitatPr-ApplianceRepair\\_6042879](https://sxa.storefront.com/shop/Appliances_appliances/shop/Appliances_appliances/HabitatPr-ApplianceRepair_6042879)). Because the page uses the same renderings, Sitecore displays the cached details for product A.

If the Vary by URL setting is selected, the cache is updated when the page for product B is called.

## Default caching settings for SXA renderings in the SXA Storefront

The default caching settings for the (not Commerce-specific) SXA renderings that are used in the SXA Storefront are set on the components that use the renderings.

The Carousel rendering is used on the SXA Storefront Home page and the Logo rendering is used in the header partial design for the SXA Storefront.

The SXA renderings that are used in the SXA Storefront are configured as Cacheable, and have the following default caching settings:

<b>Rendering</b>	<b>Clear on Index Update</b>	<b>Vary By URL</b>	<b>Vary By Currency</b>	<b>Vary By Login</b>	<b>Vary By Query String</b>
Carousel				x	
Logo	x				

## SXA Storefront architecture

The SXA Storefront is built with best practices in mind and based on modular architecture; for more information see the [Sitecore Helix development guidelines](#).

The following technologies are used in the SXA storefront:

- Sitecore Experience Platform
- Sitecore Experience Accelerator
- Basic Bootstrap
- JQuery
- Knockout – a JavaScript library that helps you create responsive user interfaces with a clean underlying data model
- Castle Windsor – a best-of-breed mature Inversion of Control container available for .NET
- Sitecore and Microsoft ASP.NET MVC
- HTML 5
- Newtonsoft JSON
- WebActivator

The following are notes on the architecture of the SXA Storefront:

- There is a clear separation of content, presentation, and catalog data.
- The storefront implementation favors standard functionality over injecting custom code and hooking into Sitecore.
- No cookies are introduced in addition to what Sitecore demands.
- The product catalog appears in Sitecore as regular items using a highly optimized data provider.
- The implementation is completely Sitecore MVC-based and follows Sitecore's MVC-first strategy. All renderings are controller based.

All components are updated using component API calls (AJAX), which has the following advantages:

- Objects and data are loaded on demand (lazy loading) and there are no page post-backs.

- There are progress indicators on buttons and tables when an operation is in process.
- Knockout.js is used for dynamic data binding.
- Component API calls return JSON.