Sitecore Experience Commerce Upgrade Guide

Sitecore Experience Commerce 9.2

August 9, 2019

Table of Contents

1. Introduction

This guide describes how to upgrade a Sitecore Experience Commerce™ (XC) 9.1 deployment to Sitecore Experience Commerce (XC) 9.2.

> **NOTE**
> These instructions apply to on-premise deployments.

1.1. Overview of the Sitecore XC upgrade process

The instructions in this document comprise the upgrade steps including Sitecore Experience Commerce (XC) and the Sitecore Experience Accelerator (SXA) Storefront .

Follow the upgrade steps in this order:

1. Review upgrade prerequisites
2. Prepare XC files for Sitecore Experience Platform upgrade
3. Upgrade Sitecore Experience Platform (XP)
4. Upgrade Commerce Engine, Connect Core and Commerce Engine Connect
5. Upgrade Commerce Business Tools
6. Upgrade SXA
7. Upgrade the SXA Storefront

2. Prerequisites

> **IMPORTANT**
> Before you begin the upgrade process, you must backup your Commerce Engine.

These upgrade instructions assume you are upgrading from Sitecore XC 9.1 to Sitecore XC 9.2.

These instructions assume that your current Sitecore XC deployment (before the upgrade) uses the following software:

- Sitecore Experience Platform (XP) 9.1.1
- Sitecore Experience Accelerator (SXA) 1.8.1
- Sitecore Experience Commerce (XC) 9.1

2.1. Install Redis

Sitecore XC 9.2 introduces distributed caching for scaled deployments, and requires an instance of Redis. Download and install the latest version of Redis for your Sitecore XC 9.2 deployment, available here.

> **NOTE**
> Sitecore recommends using Redis for Linux for production deployments. The version for Windows should only be used for development or non-production, single machine deployments.

2.2. Download and extract required software packages

In preparation for the upgrade, download and extract the required Sitecore XC 9.2 software line-up:

- Sitecore Experience Platform (XP) 9.2 software and installation instructions
- Sitecore XC 9.2 release package
- Sitecore Experience Accelerator (SXA) 1.9.0

3. Preparing for the solution upgrade

Before you begin upgrade to Sitecore Experience Commerce 9.2, you must make modifications to some existing files.

3.1. Prepare XC files for XP upgrade

As part of preparing for the Sitecore Experience Platform (XP) upgrade, the *Sitecore Experience Platform Upgrade Guide* states that you must disable Sitecore modules used in your Sitecore deployment. This section provides details that apply specifically to preparing Sitecore XC files for your XP upgrade.

Perform the following procedure *before* you perform the Sitecore XP upgrade.

To prepare the Sitecore XC files for Sitecore XP upgrade:

1. Navigate to the `App_Config/Include` directory, for example `c:\inetpub\wwwroot \<site_name>\app_config\Include\`, and remove all `Sitecore.Commerce.*.config` files from the folder and from its sub-folders (for example, move them to a different location of your choice for further reference).

2. Navigate to the `\bin` folder and delete all `Sitecore.Commerce.*.dll` files.

3. Navigate to the `\ceconnectbin` folder, and delete all `*.dll` files.

4. Remove any custom Commerce code or custom `.dll` files to avoid conflicts with breaking changes included in this release.

4. Upgrade Sitecore XP

After you have finished preparing the Sitecore XC files, you can proceed with the Sitecore Experience Platform (XP) 9.2 upgrade.

Proceed with the Sitecore XP 9.2 software upgrade by following instructions provided in the *Sitecore Experience Platform Upgrade Guide*, available on the Sitecore Experience Platform 9.2 download page.

5. Upgrade Commerce Engine, Connect Core, and Commerce Engine Connect

You must update the Commerce Engine binaries as well as policy configuration when you upgrade the Commerce Engine. You must also upgrade the Connect Core and Commerce Engine Connect files.

Perform the following procedure in the order listed:

- Update the Commerce Engine binaries
- Update policy set configuration
- Update environment configuration
- Update global policy configuration
- Update Commerce Engine configuration
- Disable antiforgery protection (temporary)
- Upgrade the database schema
- Update Commerce Solr schema
- Restart the Commerce Engine sites
- Bootstrap the Commerce Engine
- Update Commerce Connect Core
- Update Commerce Connect XConnect configuration
- Update Commerce Connect Core additional packages
- Update Commerce Engine Connect
- Synchronize Control Panel content items
- Rebuild Sitecore XC indexes
- Update the data templates
- Re-enable the anti-forgery setting
- Flush the Redis cache

5.1. Update the Commerce Engine binaries

On your Sitecore XC 9.1 system:

1. Open Internet Information Services (IIS) Manager, and stop the following Commerce Engine sites:

   - *CommerceAuthoring*

   - *CommerceMinions*

   - *CommerceOps*

   - *CommerceShops*

2. In IIS Manager, stop the *CommerceMinions_SC9* application pool.

3. In each of the following Commerce Engine service folders, replace all existing files and folders, <u>except</u> the `wwwroot` folder, with the Sitecore XC 9.2 version of those files.

   > **NOTE**
   > The Sitecore XC 9.2 files are available in the
   > `Sitecore.Commerce.Engine.*.zip` package or the version you compiled
   > from the `Sitecore.Commerce.Engine.SDK.*.zip` package.

   - `c:\inetpub\wwwroot\CommerceAuthoring_Sc9`

   - `c:\inetpub\wwwroot\CommerceShops_Sc9`

   - `c:\inetpub\wwwroot\CommerceMinions_Sc9`

   - `c:\inetpub\wwwroot\CommerceOps_Sc9`

5.2. Update policy set configuration

Sitecore XC 9.2 includes changes to Sitecore XC policy configuration, and you must update the policy set configuration files as part of the upgrade process.

You must complete these steps for each Commerce Engine service.

To update policy set configuration:

1. If you did not customize your Sitecore XC 9.1 policy set files, replace all policy set files in the `c:\inetpub\wwwroot\<CommerceEnvironment>\wwwroot\data\Environments` folder with the Sitecore XC 9.2 version of those files.

    > **IMPORTANT**
    >
    > If you have customized any of your Sitecore XC 9.1 policy files in your Commerce deployment, you must copy your changes to the Sitecore XC 9.2 version of those policy set files.

2. Delete the following Sitecore XC 9.1 policy set files, as they are deprecated for Sitecore XC 9.2:

    - `Plugin.AuthoringMemoryCache.PolicySet.*.json`

    - `Plugin.Minions.MemoryCache.PolicySet.*.json`

    - `Plugin.ShopsMemoryCache.PolicySet.*.json`

    The above listed policy set files are replaced by a new `Plugin.Caching.PolicySet.*.json` file.

The following list describes the changes to the policy set configuration files from Sitecore XC 9.1 to Sitecore XC 9.2:

- All `Plugin.*.PolicySet.*.json` files now contain a property called `UnqiueId` (populated automatically during the bootstrap process).

- The minion policies for the Node Heartbeat Minion and the Settle Sales Acrivity Minion (both deprecated in Sitecore XC 9.2) are removed from the `Plugin.Minions.PolicySet.*.json` file.

- The Indexable policy for CatalogItemsScope, CustomersScope, and OrdersScope in the `Plugin.Search.PolicySet.*.json` file now includes a property called `EntityUniqueId`.

- The Solr Search policy in the `Plugin.Search.Solr.PolicySet.*.json` file now includes properties for Solr basic authentication.

- The `PlugIn.SQL.Sharding.PolicySet.*.json` file is completely modified to support the new database structure.

- Each Sitecore.Commerce.Plugin.EntityVersions.VersioningPolicy in the `Plugin.Versioning.PolicySet.*.json` file was deprecated and replaced with a new Sitecore.Commerce.Core.VersioningPolicy. Remove each existing Versioning policy and replace with the new Versioning policy for each entity (Catalog, Category, SellableItem, and LocalizationEntity).

5.3. Update environment configuration

Sitecore XC 9.2 includes changes to the default environment configuration files.

You must complete these steps for each Commerce Engine service.

In Sitecore XC 9.2, all environment JSON files include a new "UniqueId" property. The property is added to the file automatically during the Commerce Engine Bootstrap process The following additional changes were made to the default sample environments.

5.3.1. Update the Authoring environment configuration file
Follow these steps to update the Authoring environment configuration file:

1. Open the Authoring environment configuration file (for example,
    `Plugin.Habitat.CommerceAuthoring.*.json`).

2. Find the PolicySetPolicy with a `PolicySetId` of `"Entity-PolicySet-AuthoringMemoryCachePolicySet"` and change the PolicySetId to "`Entity-PolicySet-CachingPolicySet`".

3. Remove the `ProfilesSqlPolicy` section.

4. Remove the `ProfilePropertiesMappingPolicy` section.

5. Remove the MinionPolicy with the `FullyQualifiedName` of
    `"Sitecore.Commerce.Core.NodeHeartBeatMinion, Sitecore.Commerce.Core"`.

6. Save your changes.

5.3.2. Update the Shops environment configuration file
Follow these steps to update the Shops environment configuration file:

1. Open the Shops environment configuration file (for example,
    `Plugin.Habitat.CommerceShops.*.json`).

2. Find the PolicySetPolicy with a `PolicySetId` of `"Entity-PolicySet-AuthoringMemoryCachePolicySet"` and change the PolicySetId to "`Entity-PolicySet-CachingPolicySet`".

3. Remove the MinionPolicy with the `FullyQualifiedName` of
    `"Sitecore.Commerce.Core.NodeHeartBeatMinion, Sitecore.Commerce.Core"`.

4. Save your changes.

5.3.3. Update the Minions environment configuration file
Follow these steps to update the Minions environment configuration file:

1. Open the Minions environment configuration file (for example,
    `Plugin.Habitat.CommerceMinions.*.json`).

2. Find the PolicySetPolicy with a `PolicySetId` of `"Entity-PolicySet-AuthoringMemoryCachePolicySet"` and change the PolicySetId to "`Entity-PolicySet-CachingPolicySet`".

3. In the list of Policies, add the following `PolicySetPolicy` to the collection:

```
{
    "$type": "Sitecore.Commerce.Core.PolicySetPolicy,
Sitecore.Commerce.Core",
"PolicySetId":  "Entity-PolicySet-VersioningPolicySet"
},
```

4. Save your changes.

5.4. Update global policy configuration

Sitecore XC 9.2 includes changes to Sitecore XC global policy configuration. You must update the global environment configuration file to include the policy changes as described in this topic.

You must complete these steps for each Commerce Engine service.

To update global policy configuration:

1. Navigate to the `wwwroot\bootstrap\` folder for the Commerce Engine service (for example, `c:\inetpub\wwwroot\CommerceAuthoring\wwwroot\bootstrap`).

2. Open the `Global.json` file.

3. Delete the `Components` collection.

4. Delete the following properties:

   - `DateCreated`

   - `DateDeleted`

   - `DateUpdated`

   - `EntityProviderName`

   - `IndexProviderName`

   - `IsDeleted`

5. Delete the `ListMemberships` collection.

6. Add the following new Management Cache policy to the `Policies` collection:

   ```
   {
       "$type": "Sitecore.Commerce.Plugin.Management.ManagementCachePolicy,
   Sitecore.Commerce.Plugin.Management",
       "CurrencySetsCacheName": "CurrencySets",
       "FulfillmentCacheName": "Fulfillment",
       "PaymentsCacheName": "Payments",
       "CountriesCacheName": "Countries",
       "ItemsCollectionCacheName": "ItemsCollection",
       "ItemsCacheName": "Items",
       "Expiration": 86400000,
       "AllowCaching": true
   },
   ```

7. In the `Sitecore.Commerce.Plugin.SQL.EntityStoreSqlPolicy` policy, change the value of the `"Version"` property to `"9.2.0"`.

8. Save your changes.

5.5. Update Commerce Engine configuration

Sitecore XC 9.2 includes changes to Sitecore XC Commerce Engine configuration file. You must update the `config.json` file as part of the upgrade process.

Follow these steps to update the Commerce Engine configuration file:

1. Navigate to the `c:\inetpub\wwwroot\<CE-service>\wwwroot` folder and open the `config.json` file.

2. Add the following section for caching configuration (after the `"Certificates"` section):

```
"Caching": {
    "Memory": {
        "Enabled": false,
        "CacheStoreName": "Commerce-Memory-Store"
    },
    "Redis": {
        "Enabled": true,
        "CacheStoreName": "Commerce-Redis-Store",
        "Options": {
            "Configuration": "localhost",
            "InstanceName": "Redis"
        }
    }
},
```

> **NOTE**
>
> If you are using Redis to host your cache store, you must set the `"Configuration"` and `"InstanceName"` properties to point to your Redis configuration. For more information on caching configuration, see the Commerce Engine caching topic.

3. Add the following section for compression configuration:

```
"Compression": {
    "Enabled": true,
    "ResponseCompressionOptions": {
        "EnableForHttps": true,
        "MimeTypes": [
            "text/plain",
            "text/css",
            "application/javascript",
            "text/html",
            "application/xml",
            "text/xml",
            "application/json",
            "text/json",
            "application/wasm"
        ]
    },
    "GzipCompressionProviderOptions": {
        "Level": "Fastest"
    }
}
```

4. Save your changes.

5.6. Disable antiforgery protection (temporary)

You must disable antiforgery setting for the Business Tools in the Authoring service configuration file.

> **NOTE**
> When you place a call to the Commerce Engine API from outside the Commerce Business Tools (for example, using Postman), you must disable the antiforgery protection setting in the `wwwroot\config.json` file.

To disable the antiforgery protection:

1. Navigate to the `\inetpub\wwwroot\CommerceAuthoring_Sc9\wwwroot\` folder and open the `config.json` file.

2. Locate the the anti-forgery setting and change the value to false:

```
"AntiForgeryEnabled": false,
```

5.7. Upgrade the database schema

This release of Sitecore Experience Commerce contains changes to SQL database schema and stored procedures.

> **NOTE**
> The upgrade scripts included in the `Sitecore.Commerce.Engine.SDK.*.zip` use default global and shared environment database names. Before you run the scripts, you must modify them to match the names of databases in your deployment.

To upgrade Sitecore XC SQL database schema with the changes contained in this release:

1. Unzip the `Sitecore.Commerce.Engine.SDK.*.zip` package and open the *scripts* folder.

2. Open the `SitecoreCommerce9.2.0_SchemaUpgrade.ps1` script and update the configuration parameters to match your deployment.

    > **NOTE**
    > This script applies to the Sitecore XC 9.2.0 schema in the Global and Shared Environments databases, on top of an existing 9.1.x database. The 9.1.x and 9.2.0 schemas can exist side-by-side.
    >
    > The dacpac referenced in the script is available in the on-premises WDP package, or in the Commerce Engine SDK. You can use the same dacpac for both the Global and the Shared Environments databases.

3. Launch PowerShell and run the modified `SitecoreCommerce9.2.0_SchemaUpgrade.ps1` script.

4. Open the SQL Server Management application.

5. Navigate to the unzipped `Sitecore.Commerce.Engine.SDK.*.zip` package, open the *scripts* folder and run the following scripts in the order listed:

    a. Run the `SitecoreCommerce9.2.0_DataMigration_Global.sql` script against the Global database to migrate data from Sitecore XC 9.1.x to 9.2.0.

    b. Run the `SitecoreCommerce9.2.0_DataMigration_SharedEnvironments.sql` script against the Shared Environments database to migrate data from Sitecore XC 9.1.x to 9.2.0.

> **NOTE**
>
> The script only migrates localization and versioning data for entities that have localization and versioning enabled by default (that is, catalog entities and promotion entities).
>
> If you have enabled localization for any other types of entities you must run the following command:
>
> ```
> exec[sitecore_commerce_storage].[MigrateLocalizations920]
> N'<TableNameWithoutEntitiesSuffix>'
> ```
>
> If you have enabled versioning for any other types of entities you must run the following command:
>
> ```
> exec[sitecore_common_storage].[CreateVersioningEntities920]
> N'<TableNameWithoutEntitiesSuffix>
> ```

c. Run the `SitecoreCommerce9.2.0_BuildMappings.sql` script against the Shared Environments database to populate the [sitecore_commerce_storage].[Mappings] table.

d. Run the `SitecoreCommerce9.2.0_Clean.sql` script against the Global database and the Shared Environments database to clean Sitecore XC 9.1.* data.

> **NOTE**
>
> Only run this script when you are sure all the data is correct. You can wait to run the script until you have completed all of the upgrade tasks described in this document.

5.8. Update Commerce Solr schema

Sitecore XC 9.2 uses Solr version 7.5. You must ensure that Sitecore XC indexes use the Solr schemas provided in the Sitecore XC 9.2 release package.

To update the Commerce Solr schema:

1. Open the Solr folders on the node where your Sitecore XC indexes are located (the default location is `<SOLR_INSTALL_FOLDER>\server\solr`).

2. Extract the `SIF.Sitecore.Commerce.*.zip` package and open the `SolrSchemas` folder.

3. Copy every file from the `SolrSchemas` folder to the appropriate Solr Index folder configuration folder.
   For example, copy all the files in the `SolrSchemas\CatalogItems` folder to the `<SOLR_INSTALL_FOLDER>\server\solr\CatalogItemsScope\conf` folder.

   > **NOTE**
   > The above example shows default values. Specify the index names used in your system.

4. Delete the data folder for each Solr index or clean the index.

5. Repeat step 3 for all Sitecore XC indexes.

5.9. Restart the Commerce Engine sites

To restart the Commerce Engine sites:

1. Open Internet Information Services (IIS) Manager and restart the following Commerce Engine sites:

    - *CommerceAuthoring*
    - *CommerceMinions*
    - *CommerceOps*
    - *CommerceShops*

2. In IIS Manager, restart the CommerceMinions_Sc9 application pool.

5.10. Bootstrap the Commerce Engine

You must bootstrap the Commerce Engine to propagate the changes you made to the environment configuration files into the global database.

The Sitecore XC SDK includes code samples for DevOps operations, so that you can access the Sitecore XC API directly. The following instructions assume that you are using Postman to exercise the Sitecore XC API and that you have imported the Adventure Works and Habitat environments into Postman.

To run the bootstrap operation:

1. Launch the Postman application.

2. In the **Collections** pane in Postman, navigate to the *Authentication* folder.

3. Open the `Sitecore` folder and execute the `GetToken` request.
   When Postman displays an access token in the **Body** pane, authentication is successful.

4. In the **Collections** pane, navigate to the *SitecoreCommerce_DevOps* folder.

5. Open the *1 Environment Bootstrap* folder, and execute the `Bootstrap Sitecore Commerce` request.

6. Launch Internet Information Services (IIS) Manager and perform an IIS reset.

5.11. Update Commerce Connect Core

You use the Sitecore Installer to update the Commerce Connect Core.

To update Commerce Connect Core:

1. Launch the Sitecore Installer and install the Sitecore Commerce Connect Core 13.0.*.zip package.

2. When you are prompted to overwrite existing files, click **Yes to all**. When prompted to overwrite items, select **Merge** and **Clear**, and click **Apply to all**.

5.12. Update Commerce Connect XConnect configuration

To update the Commerce Connect XConnect configuration:

1. Navigate to the `\inetpub\wwwroot\<siteName>\XConnectModels\` folder and make a copy of the `Sitecore.Commerce.Connect.XConnect.Models.json` file.

2. Paste the copied file to the following locations (overwriting the existing file):

   - `\inetpub\wwwroot\<siteName>.xconnect\App_data\jobs\continuous\IndexWorker\App_data\Models`

   - `\inetpub\wwwroot\<siteName>.xconnect\App_data\Models`

3. Open Internet Information Service (IIS) Manager and perform an IIS reset.

5.13. Update Commerce Connect Core additional packages

You use the Sitecore Installer to update the additional packages delivered with Commerce Connect Core.

To update Commerce Connect Core additional packages:

1. Stop the Marketing Automation Engine service.

2. Use the Sitecore Installer to install the following packages (if you are prompted to overwrite existing files, click **Yes to all**. If prompted to overwrite items, select **Merge** and **Clear**, and click **Apply to all**):

    -   Sitecore Commerce ExperienceProfile Core 13.0.*.zip package

    -   Sitecore Commerce ExperienceAnalytics Core 13.0.*.zip package

    -   Sitecore Commerce Marketing Automation Core 13.0.*.zip package

3. Unzip the `Sitecore Commerce Marketing Automation for AutomationEngine 13.0.*.zip` package and place the contents in the xConnect installation folder (for example, `c:\inetpub\wwwroot\xp0.connect`).

4. Start the Marketing Automation Engine service.

5.13.1. Redeploy marketing definitions

After you update the Commerce Connect Core additional packages, you must update redeploy marketing definitions in Sitecore Control Panel.

To redeploy marketing definitions:

1. Log in to Sitecore and open the Sitecore Control Panel.

2. In the **ANALYTICS** section, click **Deploy marketing definitions**. The **Deploy marketing definitions** dialog opens.

3. Click **Deploy**.

> **NOTE**
> The deployment operation can take several minutes to complete.

5.14. Update Commerce Engine Connect

You must install the Commerce Engine Connect Update module to update Commerce Engine Connect.

To update Commerce Engine Connect:

1. Launch the Sitecore Installer and install the Sitecore Commerce Engine Connect 4.0.*****.zip package.

2. When you are prompted to overwrite existing files, click **Yes to all**. When prompted to overwrite items, select **Merge** and **Clear**, and click **Apply to all**.

3. Navigate to the `\inetpub\wwwroot\<siteName>\App_Config\Include \Y.Commerce.Engine` folder and update the new `Sitecore.Commerce.Engine.Connect.config` file with any required configuration changes and to ensure the correct URL and thumbprints are being used for connectivity from Sitecore to the Commerce Engine.

4. Navigate to the `c:\inetpub\wwwroot\<siteName>\App_Config\Include \Y.Commerce.Engine` folder and rename the following (disabled) files to enable them:

    *   `Sitecore.Commerce.Engine.DataProvider.config`

    *   `Sitecore.Commerce.Engine.Connectors.Index.Common.config`

    *   `Sitecore.Commerce.Engine.Connectors.Index.Solr.*.config` (or all `Sitecore.Commerce.Engine.Connectors.Index.Azure.*.config` files if your deployment uses the Azure search provider)

5.15. Synchronize Control Panel content items

If you previously synchronized Control Panel content items to the Commerce Engine, you must re-synchronize this content to account for additional items and changes being made within these items during the upgrade process.

If you previously synchronized Control Panel content items, perform the following steps:

1. Launch the Postman application.

2. Navigate to the *ContentAPISamples* folder in the **Collections** pane.

3. Open the *DoActions* folder and execute the `Sync Content Path` sample for each environment that has a different data store (that is, has a different `ArtifactStoreId`).

> **NOTE**
> You may need to obtain the EntityView as input for the Sync Content Path sample if you receive an error related the entity version. See the Synchronize content items topic for more information.

If you have not synchronized Control Panel content items before, perform the following steps:

1. Launch the Postman application.

2. Navigate to the *SitecoreCommerce_DevOps* folder in the **Collections** pane.

3. Open the *3 Environment Initialize* folder and execute the `Ensure\Sync default content paths` sample for each environment that has a different data store (that is, has a different `ArtifactStoreId`).
    You can use the `Check Long Running Command Status` sample to ensure that the synchronization operation completes, as the operation may take several minutes.

4. Launch Internet Information Service (IIS) Manager and perform an IIS restart.

5.16. Rebuild Sitecore XC indexes

The schema update to Solr 7.5.0 requires a rebuild of Sitecore XC indexes.

To rebuild the Sitecore XC indexes:

> **NOTE**
> Before you rebuild the Sitecore indexes, make sure that previous indexes were deleted.

1. Launch the Postman application

2. In the **Collections** pane, navigate to the *SitecoreCommerce_DevOps* folder.

3. Expand the *Minions* folder and execute the following requests:

   - `Run FullIndex Minion - Orders`

   - `Run FullIndex Minion - Customers`

   - `Run FullIndex Minion - Catalog Items`

   > **NOTE**
   > When executing minion requests, make sure thta they point to the proper URL for the Minions service.

5.17. Update the data templates

To update the data templates:

1. Log in to Sitecore Launchpad (https://<host>/sitecore) and click **Content Editor**.

2. Click the **Commerce** tab.

3. Click **Delete Data Templates**.

4. Click **Update Data Templates**.

5.18. Re-enable the anti-forgery setting

Re-enable the anti-forgery setting for the Business Tools in the Authoring service configuration file.

To enable the anti-forgery setting:

1. Navigate to the `\inetpub\wwwroot\CommerceAuthoring_Sc9\wwwroot\` folder and open the `config.json` file.

2. Locate the the anti-forgery setting and change the value to true:

    ```
    "AntiForgeryEnabled": true,
    ```

5.19. Flush the Redis cache

After you have completed all of the steps required to upgrade the Commerce Engine, flush the Redis cache as described in the Redis documentation to remove any content that may have been added to the cache during the upgrade process.

6. Upgrade Commerce Business Tools

To upgrade the Commerce Business Tools:

1. Open IIS Manager, and stop the SitecoreBizFx site.

2. Navigate to the `inetpub\wwwroot\SitecoreBizFx\assets` folder.

3. Note any customizations made to the `config.json` file.

4. Delete the contents of the `inetpub\wwwroot\SitecoreBizFx` folder.

5. Copy all of the files from the Sitecore XC 9.1 `Sitecore.BizFx.*.zip` package (or from your compiled version of the Sitecore BizFX SDK) into the `inetpub\wwwroot\SitecoreBizFx` folder.

6. Update the `config.json` file (in the `\assets` folder) with any required customizations.

7. In Internet Information Services (IIS) Manager, restart the SitecoreBizFx site.

7. Upgrade the SXA Storefront

This section provides instructions for upgrading the Sitecore Experience Accelerator (SXA) Storefront.

Sitecore Experience Accelerator 1.9.0 is a prerequisite for the SXA Storefront. If you did not install Sitecore Experience Accelerator (SXA) version 1.9.0 for Sitecore as part of the Sitecore Platform upgrade, download and install the software before you proceed with the SXA Storefront upgrade.

These instructions assume that you have completed the Sitecore XC upgrade.

7.1. SXA Storefront upgrade tasks

The following list details the tasks that you must perform to upgrade the SXA Storefront:

- Install the Storefront packages
- Run the Storefront upgrade scripts
- Republish and rebuild the Sitecore search indexes
- Verify the current version number

7.2. Install the Storefront packages

To install the SXA Storefront packages:

1. Copy the following Sitecore XC 9.2 Storefront packages that you previously downloaded to the `\inetpub\wwwroot\<siteName>\App_Data\packages` folder:

    *   Sitecore Commerce Experience Accelerator
    *   Sitecore Commerce Experience Accelerator Storefront
    *   Sitecore Commerce Experience Accelerator Storefront Themes
    *   Sitecore Commerce Experience Accelerator Habitat Catalog

2. Log in to the Sitecore Launchpad and click on **Control Panel**.

3. Under **Administration**, click **Install a package**.

4. Click **Choose package** and select the package to install, for example `Sitecore Commerce Experience Accelerator`.

    > **NOTE**
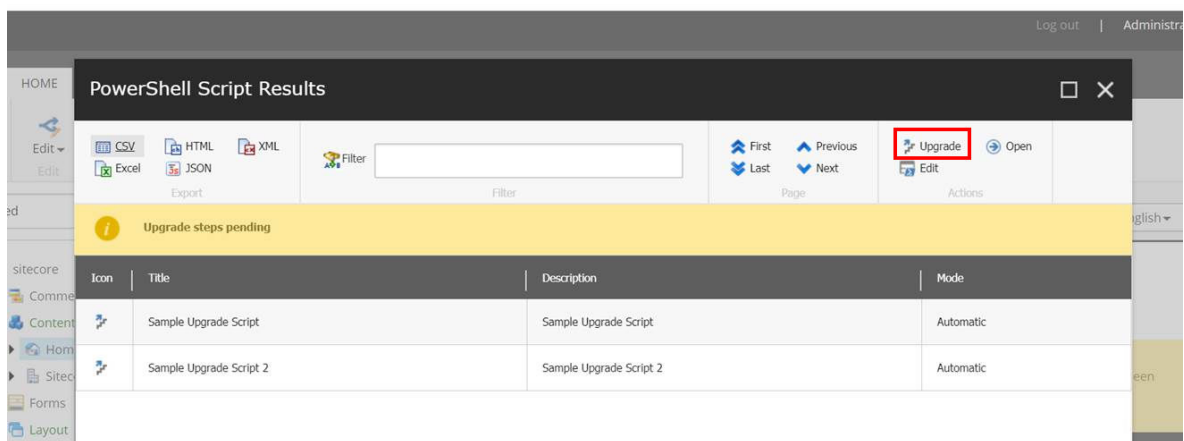   > Install the packages in the order listed in step 1.

5. Verify the package information, click **Next**, then click **Install**.

6. If you are prompted to overwrite an existing file, click **Yes to all**.

7. If the system displays a warning message that **Item being installed already exists in database**, select the **Overwrite** option and click **Apply to all**.

8. After the installation successfully completes, select the **Restart the Sitecore client** box checkbox, and click **Close** .

9. Repeat steps 3 to 8 for each Storefront package.

10. Open Internet Information Services Manager and perform an IIS reset.

7.3. Run the Storefront upgrade scripts

After you have completed the installation of the SXA Storefront package, you must run the Storefront upgrade scripts to perform the upgrade.

To upgrade the SXA Storefront:

1. In Sitecore Content Editor, navigate to the **Home** item: click **Sitecore**, **Content** and then **Home**.
   A warning displays the following message **SXA Storefront Content Upgrade Required**.

2. In the **SXA Storefront Content Upgrade Required**, dialog box, click **Upgrade**.
   The **PowerShell Script Results** dialog box opens.

3. In the **PowerShell Script Results** dialog box, in the **Actions** group on the ribbon, click **Upgrade**.

When all of the scripts have finished running, the **PowerShell Script Results** window appears empty.

7.4. Republish and rebuild the Sitecore search indexes

To republish and rebuild the search indexes:

1. Log in to Sitecore Launchpad and click **Content Editor**.

2. Click the **Publish** tab.

3. Click **Publish** and select **Publish site** from the drop-down menu.

4. In the **Publish Site** window, select **Republish – publish everything** and click **Publish**.

5. Return to the Sitecore Launchpad and click **Control Panel**.

6. Click **Indexing manager**.

7. Select the following indexes from the Local Indexes list and click **Rebuild**:

    - `sitecore_core_index`

    - `sitecore_master_index`

    - `sitecore_web_index`

8. In IIS Manager, start the upgraded Storefront site and verify that the upgrade completed successfully.

7.5. Verify the current version number

1. Log in to Sitecore Launchpad and click **Content Editor.**

2. In the left pane, navigate to *isitecore/system/Settings/Foundation/Commerce Experience Accelerator/Upgrade/Current*.

3. Verify that the value in the **Version** field is *3000*.