



relevant ecommerce™

# ZNODE & SITECORE CONNECTOR IMPLEMENTATION GUIDE

---

October 2015

# **Znode & Sitecore Connector Implementation Guide**

**Multifront Version 8.1.1**

PREPARED BY:

Znode LLC

234 W. Florida St. Suite #304, Milwaukee, WI 53204

[www.znode.com](http://www.znode.com)

## **Znode & Sitecore Connector Implementation Guide**

### **Overview**

To make the most of your shared content, some setup and ongoing maintenance will be required. This document provides everything you need to get the most of your use of these two tools, including:

- Installation & Integration Checklist
- Steps to take during initial setup
- Installing packages into Sitecore
- Running the data import in Sitecore

### **Installation and Integration: Checklist**

#### **Required software:**

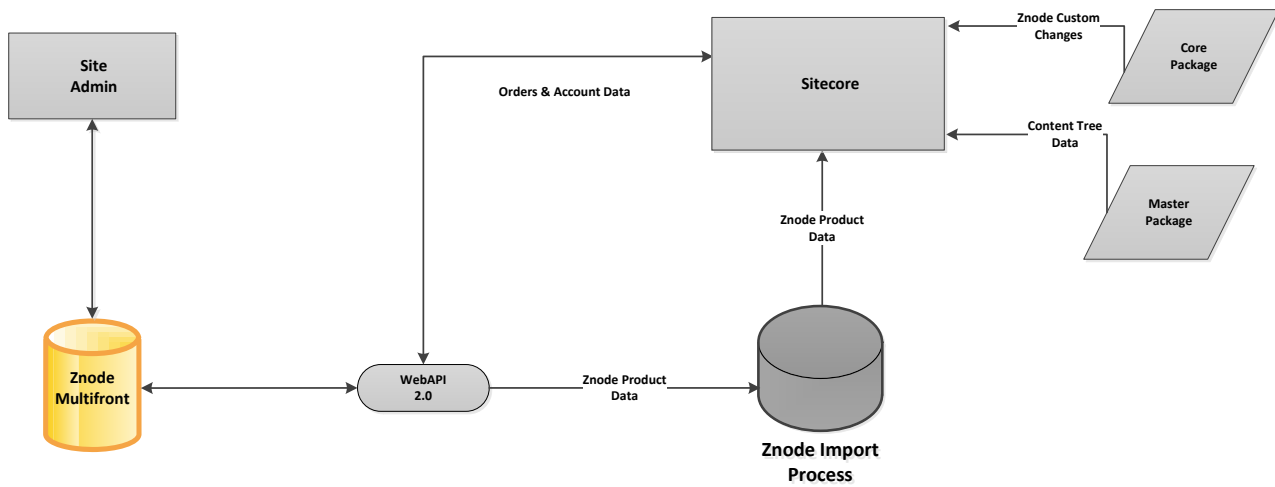
1. SQL Server 2008 R2 / 2012 / 2014
2. .Net Framework 4.5
3. IIS 7.5 & above
4. Visual Studio 2013
5. Multifront 8.0.0 Site Admin/Database
6. Znode Background Service – Configured for Lucene and index file location referenced in API web.config.
7. Sitecore from 7.0 rev. 130424

#### **Installation Steps:**

1. Run SiteCore connector installer
2. Configure and Build API
3. Deploy API to IIS
4. IIS Security Setup
5. Application Pools Setup
6. Update Multifront Domain Table
7. Install Sitecore Packages
8. Build the Znode.SiteCore.Ecommerce and the ZnodeImportProcess Solutions
9. Add the Sitecore URL to the Domain Table
10. Validate/Edit API Config File
11. Add Logging to the Sitecore Web Config File
12. Edit ZnodeHttpRequestProcessor File
13. Begin Data Import Process

## Introduction

The Znode & Sitecore connector provides customers the ability to manage product data and the presentation of content for their online store. Below is a high level diagram of the Znode/Sitecore connector:



## Run SiteCore Connector Installer

1. Run the Znode Sitecore-8.0.0-Setup.exe installer.

The default folder where all the files will be copied in the result of the installation is \ZnodeEcommerce.

**Important:** The Multifront or Marketplace installer needs to be run before the Sitecore Connector installer. This connector is additional software that integrates Multifront with Sitecore.

2. The Sitecore folder created from the installer needs to be copied into the \Multifront or \Marketplace folder.

## Configure and Build API

1. Build Znode.Multifront.Api solution from the \Znode\Znode Multifront Enterprise\ folder.
2. In Web.config, edit the <connectionStrings> section with the SQL server name, Database name and User ID and password.

```
<add name="ZNodeECommerceDB" connectionString="Data Source=localhost;Initial
Catalog=znode_multifront;user id=znodeuser;password=p@ssw0rd" />
```

## Deploy API to IIS

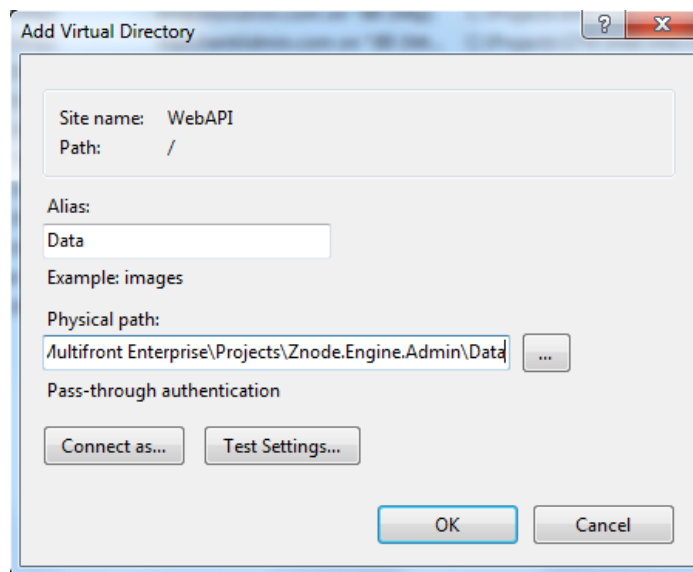
1. Open Internet Information Services Manager.
2. Right click Sites.
3. Select Add Web Site.

The screenshot shows the 'Add Web Site' dialog box with the following configuration:

- Site name:** WebApi
- Application pool:** WebApi
- Content Directory:** Physical path: .Znode Multifront Enterprise\Projects\Znode.Engine.Api
- Binding:** Type: http, IP address: All Unassigned, Port: 80, Host name: webapi.localhost.com
- Start Web site immediately:** Checked

4. Enter a Site Name.

5. Set the Physical path to the location of the Znode.Engine.Api folder.
6. Enter the IP Address for your server in the IP Address field.
7. Enter a Host name/URL.
8. Click OK.
9. Right click API site.
10. Add Virtual Directory.



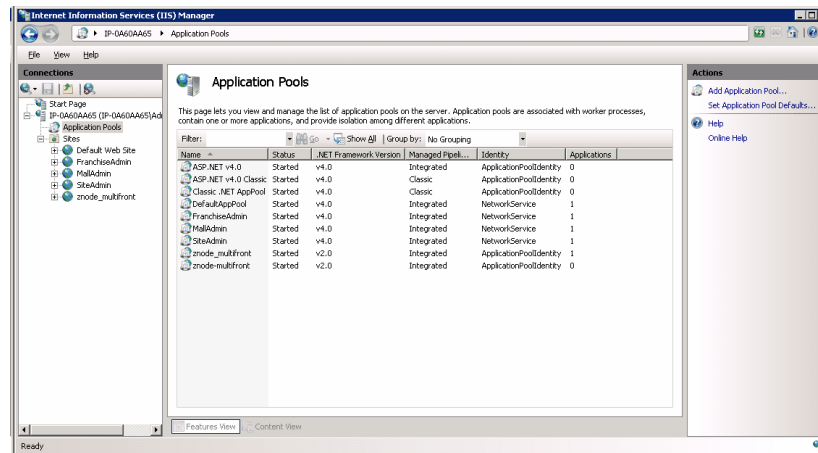
11. Set Alias to Data.
12. Set the Physical path to the location of the Znode.Engine.Admin\Data folder.
13. Click OK.

## IIS Security Setup

1. With the API website selected, click the Edit Permissions link in IIS.
2. Select the Security Tab.
3. Select the Edit button.
4. Select the Add button.
5. Enter NETWORK SERVICE into the object names box and click OK.
6. Select the NETWORK SERVICE ID if it is not chosen and click OK.
7. Edit permission to give NETWORK SERVICE Full control over the Znode.Engine.Api folder.
8. Click OK.
9. Click OK.

## Application Pools Setup

1. Click Application Pools.



2. Select the API website.
3. Select Advanced Settings.
4. In the General section, change the .NET Framework version to v4.0.
5. In the Process Model section, change the Identity to NetworkService.
6. Click OK.

**Browse the API URL. If setup properly, the Znode API landing page will appear.**

## Update Multifront Domain Table

1. Open znode\_multifront Database in SQL Server.
2. Edit/Insert a record into the domain table for the API Url created in IIS.  
(example: `INSERT INTO dbo.ZnodeDomain (PortalID, DomainName, IsActive) VALUES (1, 'API.Znode.com', 1, ApiKey, NEWID())`)
3. Commit the changes.

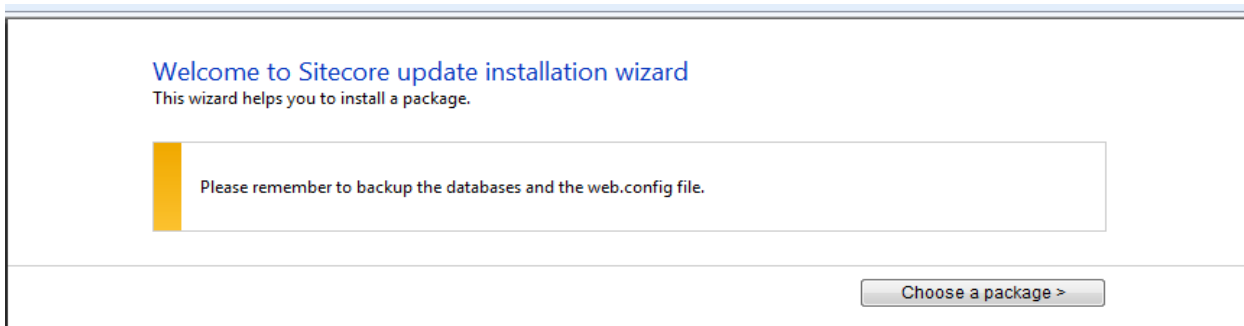
## Install Sitecore Packages

With the API being built and configured successfully, the next step is to setup the Sitecore portion. There are two files that need run by the Sitecore installation wizard. The two files are discussed below:

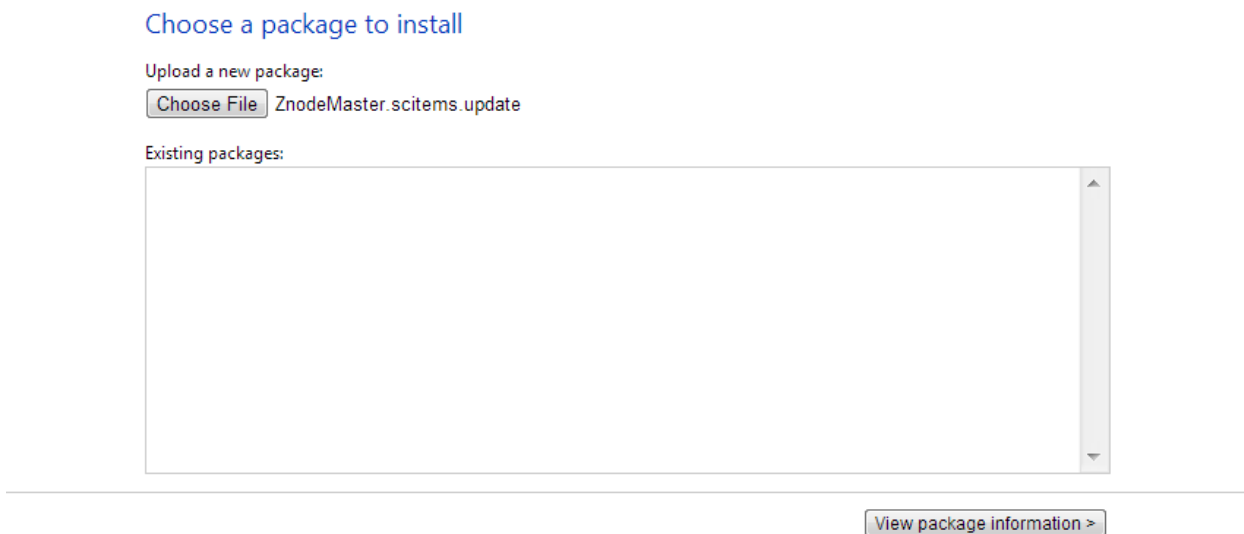
- Master File \ZnodeEcommerce\Sitecore\Packages\ZnodeMaster.scitem.update - This file will be run against the master database and creates the Sitecore/Znode content tree structure.
- Core File \ZnodeEcommerce\Sitecore\Packages\ZnodeCore.scitem.update - This file will be run against the Core Database and will install the import buttons along with the My Znode Toolbar in Sitecore.

### **To install your Sitecore packages:**

1. Navigate to the Sitecore installation wizard. In the URL string, add the following after Sitecore; admin/UpdateInstallationWizard.aspx. For example:  
`http://znodesitecore.local.com/sitecore/admin/UpdateInstallationWizard.aspx`.



2. Browse for the master package: \ZnodeEcommerce\Sitecore\Packages\ZnodeMaster.scitems.update:



3. Select '**View package information**', button.
4. Once a package is uploaded, it needs to be analyzed and installed.
  - The first screen shot allows the review of the package. When ready to proceed, select the '**Analyze and Install the package**' button.
  - The second screen shot allows the package to be analyzed and/or installed.

**Important:** The package does not have to be analyzed first before being installed.



### Review package information

You are about to install the following package.

Name: ZnodeSitecoreTDS\_20131002\_0834  
Location: /sitecore/admin/packages/znodemaster.scitem.update  
Version: 1.0  
Author: Znode Inc.  
Publisher: Znode Inc.

Readme:

< Choose a different package

Analyze and install the package >

### Analyze and install the package

You are about to install **ZnodeSitecoreTDS\_20131002\_0834**.

Files overwritten or deleted during the installation will be saved to a backup folder.



#### Analyze

Analyze the package for potential installation conflicts



#### Install

Install the package immediately

< Choose a different package

5. Select the Install button (see previous screen shot).
6. Confirm the installation of the package is successful as seen below:
  - The first screen displays the progress bar of the installation. When the installation is complete, select the '**Installation result**' button.
  - The second screen shot displays the successful message.

## Installing the update package

Processing action 158 of 220



▼ [More information](#)

[Installation result >](#)

## The package has been installed

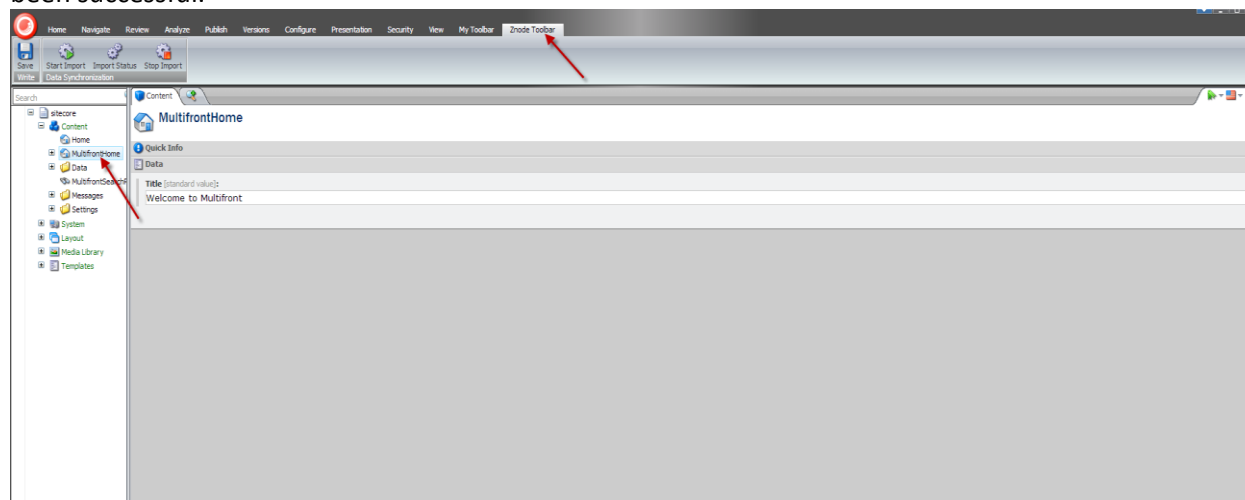
15 potential problems were encountered during the installation, including 15 collisions.

▼ [Show installation messages](#)

[< Install another package](#)

7. Repeat package installation steps for the core package.  
`\ZnodeEcommerce\Sitecore\Packages\ZnodeCore.scitems.update.`
8. Navigate to the Content Editor Section of Sitecore and confirm the toolbar and content tree have been customized.

**Important:** If the Znode Toolbar appears with the Import buttons, Sitecore core package installation has been successful.



## Build the Znode.SiteCore.Ecommerce and the ZnodeImportProcess Solutions

1. Build the Znode.SiteCore.Ecommerce solution from  
\ZnodeEcommerce\Sitecore\Znode.SiteCore.Ecommerce folder.
2. Build the ZnodeImportProcess solution from\ZnodeEcommerce\Sitecore\ZnodeImportProcess folder.
3. Copy Znode Ecommerce and Znode Import Process files to Sitecore website.
  - a. In order to copy the above files, the Install Multifront Controls.bat script can be run which is located in the ZnodeEcommerce\Sitecore folder.

## Add the Sitecore URL to the Domain Table

In order to render the Znode/Sitecore site, the URL associated to your Sitecore instance will need to be added to the Znode Domain table. This will be required for all stores. This can be done by performing the following steps:

### **To add the URL to the Domain Table:**

1. Open SQL Server.
2. Go to the ZnodeDomain table in znode\_multifront database.
3. Right click and select Edit Top 200 rows.
4. Edit/Insert a record into the domain table for your Sitecore URL in IIS.

(example: **INSERT INTO** dbo.ZnodeDomain (PortalID, DomainName, IsActive)**VALUES** (1, **Sitecore.Znode.com'**, 1, ApiKey, NEWID())

5. Commit the changes.

## Validate/Edit API Config File

1. From \SitecoreWebsiteFolder\Website\App\_Config\Include, validate the ZnodeWebAPI.config file URL and Key are correct. The first line will be for the API. Each subsequent line will be for each store, where setting name = "StoreURL" and value = "APIKey". Example:
  - a. <setting name="WebAPIUrl" value="webapi.localhost.com"/>
  - b. <settingname="localhost" value="SD2B4CSE-D8B3-4488-904D-64094762E136"/>
  - c. <settingname="localhost/b2b" value="7ACC9817-F4FE-43D0-81CB-C7551B1SFS1C"/>
2. You can validate the URL by going into IIS and select edit bindings from the store site.
3. You can validate the domain key by the following:
  - a. Open SQL Server.
  - b. Go to the ZnodeDomain table in znode\_multifront database.
  - c. Validate the domain key for the store site.

YBLAPFY7LMQ1\S...bo.ZNodeDomain					
	DomainID	PortalID	DomainName	IsActive	ApiKey
▶	1	1	localhost	True	5D2B4C5E-D8B3-4488-904D-64094762E136
	2	6	localhost/b2b	True	7ACC9817-F4FE-43D0-81CB-C7551B15F51C
	3	5	localhost/Specialty	True	958C8161-617E-40D0-B405-08B13633A0E8
	7	1	localhost/web	True	4FB7C166-AF0F-43A2-A02A-EFB14D5ADAFF
	8	1	localhost/api/v1	True	0F39E5C4-5925-491A-8714-B2A6E3D164C2
	9	1	webapi.localhost.com	True	0F39E5C4-5925-491A-8714-B2A6E3D164C2
	11	1	znodesitecore.local.com	True	0F39E5C4-5925-491A-8714-B2A6E3D164C2
	12	1	siteadmin.znodesitecore.com	True	0F39E5C4-5925-491A-8714-B2A6E3D164C2
*	NULL	NULL	NULL	NULL	NULL

## Add Logging to the Sitecore Web Config file

Add the following code which will incorporate extra logging for troubleshooting.

- Add the following code to the <log4net> section to the web.config file located in \SitecoreWebsiteFolder\Website.

```
<appender name="ZnodeSitecoreImportFileAppender"
type="log4net.Appender.SitecoreLogFileAppender, Sitecore.Logging">
  <file value="$(dataFolder)/logs/ZnodeSitecoreImport.log.{date}.txt"/>
  <appendToFile value="true"/>
  <layout type="log4net.Layout.PatternLayout">
    <conversionPattern value="%4t %d{ABSOLUTE} %-5p %m%n"/>
  </layout>
</appender>
<logger name="Znode.Sitecore.Diagnostics.Import" additivity="false">
  <level value="INFO"/>
  <appender-ref ref="ZnodeSitecoreImportFileAppender"/>
</logger>
```

## Edit ZnodeHttpRequestProcessor File

- From the \SitecoreWebsiteFolder\Website\App\_Config\Include folder open the ZnodeHttpRequestProcessor config file.
- Locate the sites section and modify the hostName field which needs to match the domain of the store site. Both example sites will need to be configured in the ZnodeWebAPI.config and ZnodeHttpRequestProcessor.config.

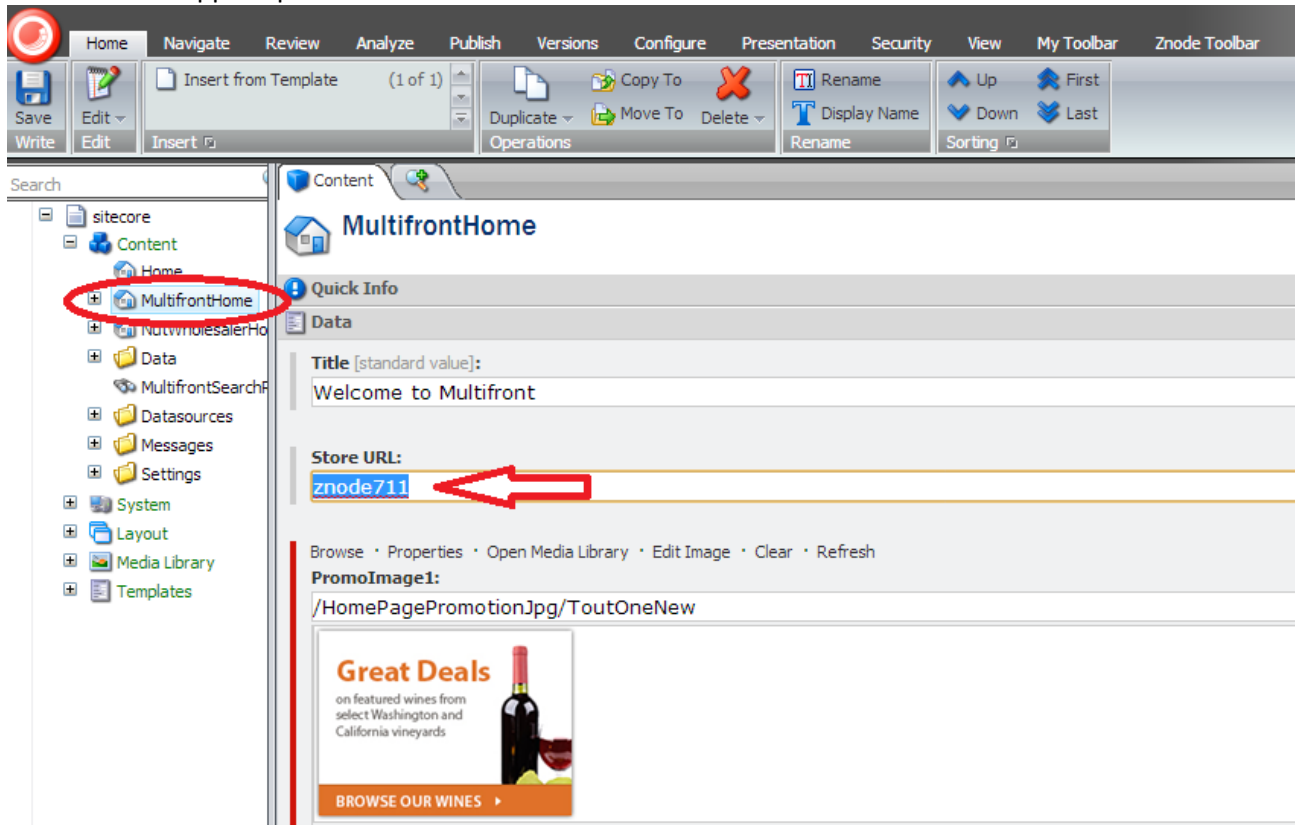
```
<sites>
  <site name="multifrontsite" hostName="multifrontsitecore7.com" virtualFolder="/" physicalFolder="/" content="master"
  rootPath="/sitecore/content" startItem="/MultifrontHome" database="web" domain="extranet" allowDebug="true" cacheHtml="false"
  htmlCacheSize="10MB" registryCacheSize="0" ViewStateCacheSize="0" xslCacheSize="5MB" filteredItemsCacheSize="2MB"
  enablePreview="true" enableWebEdit="true" enableDebugger="true" disableClientData="false" patch:after="site
  [name='modules_website']"/>
  <site name="nutwholesaler" hostName="mysecondstore.com" virtualFolder="/" physicalFolder="/" content="master"
  rootPath="/sitecore/content" startItem="/NutWholesale" database="web" domain="extranet" allowDebug="true" cacheHtml="false"
  htmlCacheSize="10MB" registryCacheSize="0" ViewStateCacheSize="0" xslCacheSize="5MB" filteredItemsCacheSize="2MB"
  enablePreview="true" enableWebEdit="true" enableDebugger="true" disableClientData="false" patch:after="site
  [name='modules_website']"/>
</sites>
```

Change to match  
domain url

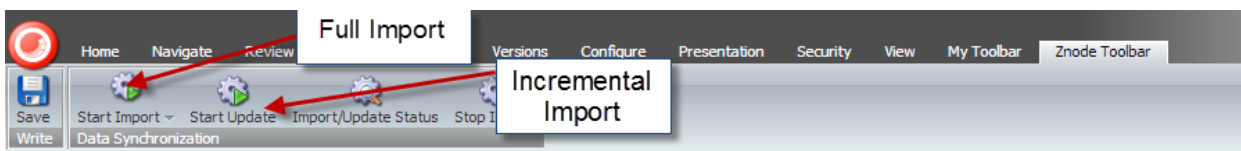
## Begin Data Import Process

To begin the import process in Sitecore:

1. Confirm the Store URL for each store. These should match the store's Start Item as mapped in ZnodeHpptRequestProcessor:



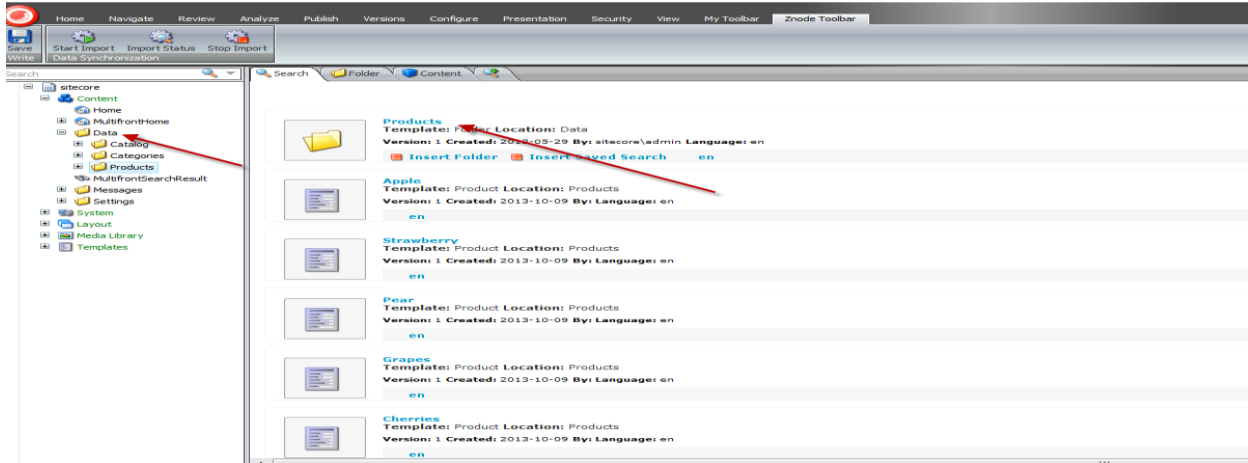
2. Navigate to the Content Editor section of Sitecore.
3. Select the 'Znode Toolbar' navigation bar. The following options are available:
  - a. Full Import:
    - Ability to import data for all stores that are in Sitecore
    - Ability to import data for one store in Sitecore
  - b. Incremental Import:
    - Ability to import incremental updates for all stores that are in Sitecore
    - Ability to import incremental updates for one store in Sitecore



4. Choose if a full or incremental import needs performed. In an initial implementation, a full import would be selected.

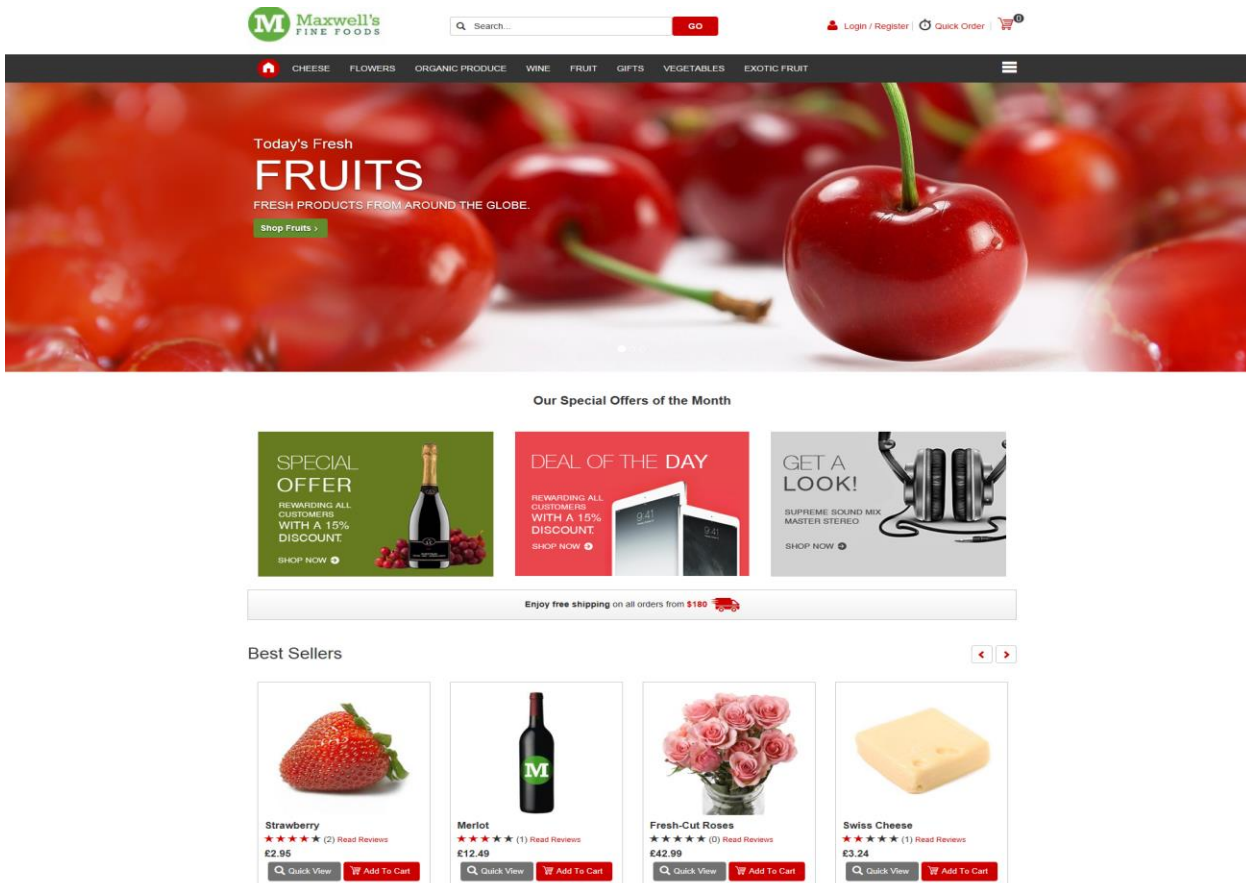


5. Validate the import has completed by selecting the 'Import Status' button.
  - a. Validation of the data can be performed by expanding the data item and selecting the catalog, category and/or product items in the content tree. Screen shot below:



Lastly select Preview in Content Editor or enter the store URL to validate the storefront renders as expected.

\*Sitecore Preview mode will not show type ahead in search bar. This will require a published version of the site running in current browser.



## Appendix A

Below is the list of files that are copied when the “Install Multifront Controls.bat” script is run.

### Znode Sitecore Ecommerce files:

1. **Copy DLL's in Bin folder –**
  - a. From - \ZnodeEcommerce\Sitecore\Znode.SiteCore.Ecommerce\Bin select all of the dll files
  - b. To - Copy to \SitecoreWebsiteFolder\Website\Bin
2. **Copy Global.asax/Suggestions files –**
  - a. From - \ZnodeEcommerce\Sitecore\Znode.SiteCore.Ecommerce copy the Global.asax, GetSuggestions.asmx.cs and GetSuggestions.asmx files
  - b. To - Copy them to the \SitecoreWebsiteFolder \Website
3. **Copy Script files –**
  - a. From – \ZnodeEcommerce\Sitecore\Znode.SiteCore.Ecommerce\Scripts copy all files
  - b. To - \SitecoreWebsiteFolder \Website \Scripts
4. **Copy Sublayout folder –**
  - a. From - \ZnodeEcommerce\Sitecore\Znode.SiteCore.Ecommerce\Layouts copy the Sublayouts folder
  - b. To - \SitecoreWebsiteFolder \Website \Layouts
5. **Copy Main.aspx file –**
  - a. From - \Znode.SiteCore.Ecommerce layouts folder copy the Main.aspx file
  - b. To - \SitecoreWebsiteFolder \Website\Layouts
6. **Copy App\_GlobalResources, Images, bootstrap folders –**
  - a. From - \ZnodeEcommerce\Sitecore\Znode.SiteCore.Ecommerce folder copy the above folders
  - b. To - \SitecoreWebsiteFolder\Website
7. **Ecommerce App Config File –**
  - a. From - \ZnodeEcommerce\Sitecore\Znode.SiteCore.Ecommerce\App\_Config\Include copy all six files
  - b. To - \SitecoreWebsiteFolder\Website \Website\App\_Config\Include

### Znode Import Process files:

1. **Copy Binaries Files –**
  - a. From - \ZnodeEcommerce\Sitecore\ZnodeImportProcess\Binaries copy all files
  - b. To - \SitecoreWebsiteFolder\Website\Bin
2. **Unity Config File –**
  - a. From -  
\ZnodeEcommerce\Sitecore\ZnodeImportProcess\Znode.Sitecore.DataImport\App\_Config  
copy the Znode.Sitecore.DataImport.Unity.config file
  - b. To - \SitecoreWebsiteFolder\Website\App\_Config
3. **Import Config File –**
  - a. From -  
\ZnodeEcommerce\Sitecore\ZnodeImportProcess\Znode.Sitecore.DataImport\App\_Config\Inc  
lude folder, copy the Znode.Sitecore.DataImport.config file
  - To - \SitecoreWebsiteFolder\Website\App\_Config\Include