System Requirements

To install Znode Storefront you need to have familiarity with Inernet Information Services (IIS), Microsoft .NET Framework and SQL Server 2005.

System Requirements (Production Server)

- Windows 2003, 2008
- IIS 6 or IIS 7
- .NET Framework 2.0 or 3.5
- .NET Framework 3.5 - Required for Znode Middleware
- SQL Server 2005 or 2008
- Enable Full Trust
- ASP.NET 2.0 AJAX Extensions 1.0 (Not required for .NET 3+)

System Requirements (Development PC)

- Visual Studio 2005 or 2008
- Windows XP, 2003, Vista, or 2008
- .NET Framework 2.0 or 3.5
- .NET Framework 3.5 - Required for Znode Middleware
- SQL Server 2005 or 2008

See Also:

Installing Znode Storefront

Development Installation

Please see the steps below to install Znode Storefront on your development environment. Note that you should check your system requirements before proceeding

Step 1: Install the Znode Storefront MSI

- After you purchase Znode Storefront, you will receive a link to download an installer (MSI)
- This installer will deploy all the Znode Storefront code and utilities to your computer
- Note that you do not need to run this MSI on a production server - you can upload the compiled files using FTP. See Production Server Installation for further instructions.

Step 2: Create the Database

- The database creation script can be found under the *$ProgramFiles$\ZnodeStorefront\$Edition$ \SQL* folder.
- Open SQL Server Enterprise Manager using an administrator account.
- Run the *znode_singlefront_database.sql* script. This will create a new database called *znode_singlefront*
- Create a SQL Login with userid = znodeuser with password = p@ssw0rd (note that you can create a different sql login or use windows authentication - in those cases you need to update the web. config appropriately) .
- Assign DBO permissions to the *znodeuser* login for the *znode_singlefront* database

Step 3: Open the Visual Studio Solution

Running Under .Net 2.0
Use these instructions if you would like to run Znode Storefront using .Net 2.0

- The Visual Studio solution can be found under the *$ProgramFiles$\ZnodeStorefront\$Edition$\Web* folder.
- Open the solution using Visual Studio 2005. If you open it using Visual Studio 2008, you need to convert the solution. **When it prompts you to make a backup say "No" or select a location outside of your Web folder**. When prompted to convert the libraries to .Net 3.5 you should select " **No**" to continue running your libraries in .Net 2.0.
- Once the solution is open, you must update the database connection settings in the web.config file.
- Compile and run the solution in Visual Studio - this will open the Znode Storefront demo web site. You should be able to navigate the demo store and try out the different storefront functionality.

Running Under .Net 3.5
Use these instructions if you would like to convert the Znode Storefront solution to use .Net 3.5

- The Visual Studio solution can be found under the *$ProgramFiles$\ZnodeStorefront\$Edition$\Web* folder.
- Download and install the Ajax Control Toolkit for .Net 3.5. from http://ajaxcontroltoolkit.codeplex.com/Release/ProjectReleases.aspx?ReleaseId=16488
- Copy the *AjaxControlToolkit.dll* from the extracted Ajax Control Toolkit for .Net 3.5 directory to the *Web/Bin* folder replacing the one that shipped with Znode Storefront.
- Copy the *web.config* file supplied in the *Maintenance/.Net 3.5* folder to your projects *Web* folder.
- Remove the *System.Web.Extensions.dll* and System.*Web.Extensions.Design.dll* files from the *Web/Bin* folder.
- Open the solution using Visual Studio 2008. You will be prompted to convert the solution. **When it prompts you to make a backup say "No" or select a location outside of your Web folder**. When prompted to convert the libraries to .Net 3.5 select "**Yes**".
- Once your project is open in Visual Studio right click on **each** library and choose "Properties". Under the "Target Framework" drop-down choose ".NET Framework 3.5".
- Remove the references to the *System.Web.Extensions.dll* and *System.Web.Extensions.Design.dll*.
- Update the database connection settings in the web.config file.
- Compile and run the solution in Visual Studio - this will open the Znode Storefront demo web site. You should be able to navigate the demo store and try out the different storefront functionality.

64 Bit Support
By default Znode Storefront is delivered to run on mixed platforms (32 bit and 64 bit). You will need to make some adjustments to your IIS settings to run Znode Storefront in compatibility mode on 64 bit systems. If you need to run in 64 bit mode on your server a special version of the base library has been provided. This library can be found in *\Maintenance\64 bit Support\ZNode.Libraries.Framework.Business.dll*. Drop this DLL into your Web\bin directory and recompile your application for a 64 bit platform.

Step 4: Administering the Storefront

- You can access the storefront management tool using *http://$YourWebsite$/admin*
- The default credentials are: UserID=admin and Password=admin - You will be prompted to change the password the first time you login.

Step 5: Diagnostics

- Browse to the *http://$YourWebsite$/diagnostics.aspx* page to run detailed diagnostics and application trace on your storefront installation.
- You can also inspect the log files that can be found under *Web\Data\Default\Logs* folder - this can help debug various issues.

Step 6: Apply Patches

- Open a browser to http://www.znode.com/support/downloads.aspx to download the latest patches for your version of the software.
- Apply the patches to your source code, recompile and test.

See Also:

Production Server Installation

When you are ready to deploy your storefront to a production server such as a shared hosting or dedicated server please follow these steps:

Step 1: Prepare the Development Storefront for Deployment

- Before deploying the storefront to production you should have completely customized, configured and tested the storefront in your development environment (see Development Installation).
- Compile your storefront in development using the "Publish" option in Visual Studio.
- Back up your development database.
- Upload the compiled storefront and database to your production server(s) using FTP and move the compiled web site to it's final destination.

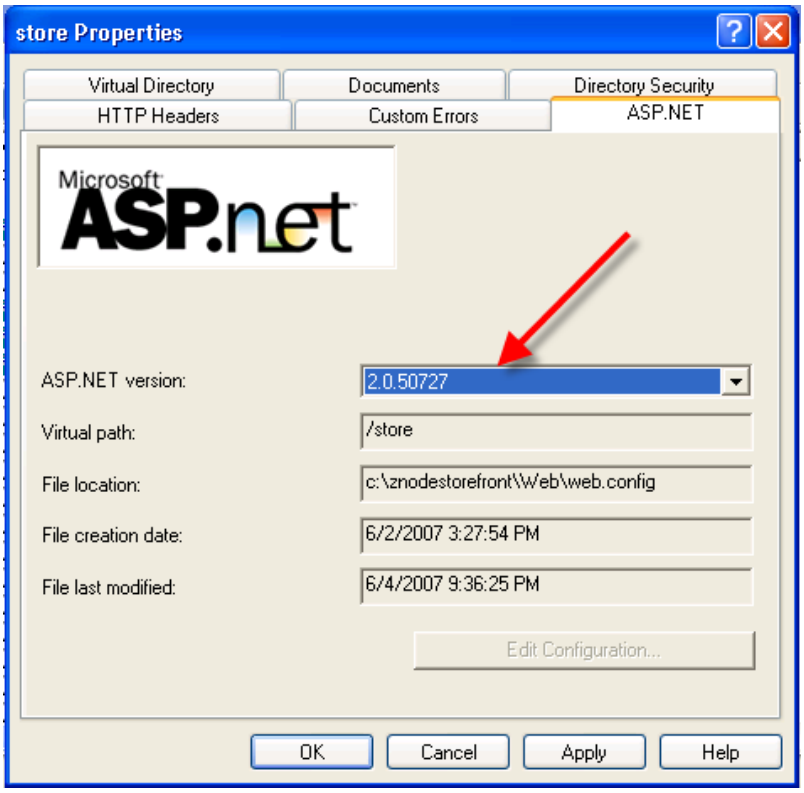
Step 2: Prepare the Storefront for Deployment

Create a new Website or Virtual Directory

- Open Internet Information Services using the Microsoft Management Console.
- Configure a new virtual directory or website in IIS.
- In the Directory setting enter the path to the "Web" folder for the compiled storefront. Click Next.
- Make sure that the Access Permissions are set to "**Read**" and "**Run scripts (such as ASP)**". Click Next.
- Click Finish in the Virtual Directory or Website Creation Wizard.

Configure the Website or Virtual Directory

- Edit the "Properties" for your Website or Virtual Directory.
- Click on the "ASP.NET" tab of the property page.
- Under "ASP.NET version" select "**2.x.xxx**". A screen shot of what this property page should look like is below.
- You may need to apply a patch for .Net 2.0 in order for menus to display correctly in IE 8. For more information see http://support.microsoft.com/kb/962351.



Step 3: Configure Permissions

Configure permissions based on your operating system as follows:

Windows 2003 or 2008 Server

The *Network Service* account should have

- Read permissions at the "Web" folder level
- Read + Write + Modify permissions at the "Web/Data" folder level

Step 4: Install the Storefront Database

- Restore the database that you uploaded to your server.
- Add a SQL User with DBO access to this new database. You can alternatively use Windows authentication if you prefer.

Step 5: Configure Web.Config Connection String

- Open the web.config file in your website under the root folder
- Edit the connection string to point to your new database. Change the data source, initial catalog, user id and password to the new settings. The connection string looks like this:

```
<connectionStrings>
     <add name="ZNodeECommerceDB" connectionString="Data Source=LOCALHOST\SQLEXPRESS;
          Initial Catalog=znodestorefront;user id=znodeuser;password=p@ssw0rd" />
</connectionStrings>
```

- If your SQL Server is configured to use Windows Authentication only then you should change the connection string to use Integrated security instead.

Step 6: Test the Site Configuration

- Open the storefront by going to *http://<your-storefront>/default.aspx*.(replace "<your-storefront>" with the domain name of your store). You will be prompted to activate a valid storefront license. After you activate a valid license you can browse the store.
- You can open the diagnostics page by going to *http://<your-storefront>/diagnostics.aspx* .
- On the diagnostics page you will clearly see if you are connecting to the database and if you have the correct permissions set on your directories. If you see an error about the SMTP service not being set up correctly don't worry. You can set this up later in the Admin.

Step 7: Configure the Site for Production

- Refer to the Znode Storefront PCI Guide for important information on configuring the storefront and your server for use in a production environment.
- In addition edit the following settings to look like the following in your web.config to disable the development diagnostic tools:

```
<!-- These settings are defaulted to work with a development environment. -->
<!-- Be sure to set each of these items before going to production. -->
<add key="EnableDebugging" value="0"/>
<add key="EnableDiagnosticsPage" value="0"/>
<add key="EnableIntegrationTest" value="0"/>
<add key="EnableActivationPage" value="0"/>
<!-- Production URL : https://gateway.fedex.com/web-services -->
<!-- Test Server URL : https://gatewaybeta.fedex.com/web-services -->
<add key="FedExGatewayURL" value="https://gateway.fedex.com/web-services"/>
```

See Also

Activating Znode Storefront

Activating Znode Storefront

This topic provides you with an overview of the Znode Storefront licensing and activation process. It is extremely important for you to understand this process in order to keep your storefront compliant with the Znode licensing requirements.

What does Activation mean?

When activating the storefront you are required to enter your license key. The storefront will then send this license key to the Znode activation server for verification. Once verified, a license file will be written to the Data/Default/Config directory. At this point your software will be licensed and the storefront will not need to contact license server again unless it is moved to another server.

The SingleFront Edition is licensed in two ways:

- Single domain, single processor - This license allows you to have one domain name running on a single processor. Domain names are fully qualified so for instance *http://MyStore.com*, *http://MyStore.com/store* and *http://test.MyStore.com* are all considered separate domains and require separate licenses.
- Multiple domain, single processor - This license allows you to run an unlimited number of domains on a single processor.

For both of these license types, if your server has multiple processors or if you use a load balanced server you will need additional licenses for each processor. "Multi-Core" processors are considered as one processor and only require one license.

When do I need to activate?

When running your storefront from within Visual Studio or accessing the site from *http://localhost* (or any variation such as *http://localhost/store*) you do not need to activate your software. This mode allows you to do all your development and local testing without any restrictions. Even though no activation is needed, development licenses may be required.

As soon as you need to access your site from a URL other than *http://localhost* you will need to activate the software. For instance accessing your site from *http://test.MyStore.com* or *http://www.MyStore.com* will require you to activate.

How many times can I activate?

For the single domain license you can activate as many times as needed on a single domain. Once activated, you can not activate the storefront again to a different domain. You may not run the storefront on a load balanced server without purchasing additional licenses for each processor.

With the multiple domain license you can activate as many times as needed on a single processor. Each activation can use a different domain if you like. Once activated you can not activate on another processor.