# Contents

1  **GETTING YOUR WEBSITE ONLINE** ................................................................................. 3  
1.1 Publishing Using WebMatrix  ..................................................................................... 3  
1.2 Publishing to Your Web Hosting Provider via FTP  .................................................. 4  
1.3 Publishing to Your Own IIS  .................................................................................... 5  

2  **TROUBLESHOOTING** ................................................................................................. 9  
2.1 Turning On Detailed ASP.NET Error Messages  ....................................................... 9  
2.2 Wrong .NET Version  ................................................................................................. 9  
2.3 No Sufficient Permissions on the Website Directory  ............................................. 11  
2.4 No Write Permissions on the Website Directory  ................................................... 13  
2.5 Not Run as an Application on IIS  ........................................................................... 14
1 Getting Your Website Online

Once you have created a C1 CMS website locally, you normally want to make it available online, that is, publish it.

Basically, all you need is web hosting that allows you to run the ASP.NET version required by C1 CMS (see “System Requirements”) - in Full Trust mode.

Please note that not all web hosting providers offer ASP.NET services or allow running ASP.NET applications in Full Trust mode. You should check these prerequisites with your provider before publishing the website.

Publishing a C1 CMS website is pretty straightforward. You should upload the files of your C1 CMS website to your web hosting provider via FTP or using any other upload method.

Please note that if your C1 CMS website is running on WebMatrix, you may as well use WebMatrix’s publishing feature.

In this quick deployment guide we will consider three ways of publishing your C1 CMS website:

- Publishing using WebMatrix
- Publishing to your web hosting provider via FTP
- Publishing to your own IIS

Should you have any trouble while publishing your website, check the Troubleshooting section at the end of this guide.

1.1 Publishing Using WebMatrix

Being a good all-in-one platform to try a Web application such as C1 CMS, WebMatrix also allows you to publish websites using its publishing feature.

You will need publishing information from your web hosting provider to configure WebMatrix. Some providers are WebMatrix-aware and can provide you with a Publishing Settings file (.PublishingSetting), which you should import to have WebMatrix configured automatically.

Once configuration is complete, you should select the protocol to use for publishing. WebMatrix offers FTP and Web Deploy.

If you do not have web hosting yet, you can use WebMatrix to find it.
1.2 Publishing to Your Web Hosting Provider via FTP

If you own an account with a web hosting provider, you can publish your C1 CMS website via FTP (or any other method that you provider offers for deploying website).

To successfully run a C1 CMS website on the server, you should make sure with your provider that:

- your web hosting supports ASP.NET applications
- and they can run in Full Trust mode.

Uploading a website via FTP is a simple XCOPY operation:

1. Connect to your web hosting provider via FTP with your credentials using your favorite FTP client (e.g. FileZilla).
2. Copy (upload) all the folders and files to the root folder.

Once done, you can access your website online by the URL given to you by your provider.

If you run your local website on WebMatrix, consider “Publishing Using WebMatrix”.

Many providers allow you to upload your website packed as a ZIP archive, and they will unzip it on the server for you. If so, you should zip your website before uploading it. We recommend using our ad-hoc add-on for this purpose: XML-Based Site Backup.
Please note that you should copy the files and folders in the root of your local website to the root folder on the remote server. If you upload your files in a subfolder on the remote server, you would need to take additional steps to configure your remote website subfolder to work as an application provided that your hosting plan allows this.

1.3 Publishing to Your Own IIS

If you have access or own a remote server with IIS, you can publish a local C1 CMS website to this IIS. The following steps are intended for IIS version 7.5 or later.

1. Create or make sure to have an application pool with the .NET Framework version required by C1 CMS (see “System Requirements”).
2. We highly recommend setting its managed pipeline mode to “Integrated” (although C1 CMS may as well run in “Classic” mode).

Figure 2: Configuring an application pool on IIS 7 for C1 CMS

3. Copy your website files to a folder on the remote server (XCOPY).
4. For the identity the application pool is running under (for example, “NETWORK SERVICE”), on this with your website, set permissions to allow modifying, reading, writing etc.
Figure 3: Setting read and write permissions on the website directory

5. Create a website on IIS ("Add Web Site..."), selecting the application pool from Step 1 and point it to the website’s folder from Step 3.
6. If needed, specify a port other than 80 and the host name.
7. Save the configuration (OK).

If you run your C1 CMS website in a subfolder on IIS, make sure that it works as an application, and not as a Virtual Directory. You simply need to select “Add Application” in the context menu of the root website you add a C1 CMS website to.
Figure 5: Adding a subfolder as an Application

If you have already added a C1 CMS website as a virtual directory, you can still convert it to an application, selecting “Convert to Application” in the context menu.

Figure 6: Converting a Virtual Directory into an Application
2 Troubleshooting

Here you can learn about fixing common errors encountered after publishing C1 CMS websites.

To successfully troubleshoot errors, you should first turn on detailed ASP.NET error messages.

2.1 Turning On Detailed ASP.NET Error Messages

A good start to finding a fix to an error is turn on the detailed ASP.NET error messages in web.config and re-publish web.config again.

In most cases, you will access your website remotely. By default, remote users will see custom errors (“RemoteOnly”), which are not informative. You have to turn off custom errors (“Off”). This automatically enables detailed ASP.NET errors.

1. Edit web.config on the local copy of your website.
2. Locate the element: configuration/system.web/customErrors
3. Set the value of its mode attribute to “Off”.

```xml
<configuration>
  <system.web>
    <customErrors mode="Off" />  
  </system.web>
</configuration>
```

Listing 1: Turning on detailed error messages in web.config

4. Save the file.
5. Re-publish web.config to your hosting server.
6. Reopen the website.

When you start the website, you will be able to see detailed information about errors if any.

Please note that once the error is detected and fixed, you should turn off the detailed error messages, that is, turn the custom errors back on. For this, repeat Steps 1-5 but this time set the value to “RemoteOnly”.

The following is the list of typical errors you may encounter after publishing your C1 CMS website to your Web hosting.

- Wrong .NET version
- No sufficient permissions on the website directory
- No write permissions on the website directory
- Not run as an application on IIS

2.2 Wrong .NET Version

When you open your website, you can get the server error: “Unrecognized attribute ‘targetFramework’”.
How to Get Your Site Online

Figure 7: Error when running on the wrong .NET version

This error occurs if your website is running on the wrong .NET version.

If you have access to IIS, do as described below. Otherwise, follow the standard procedure of changing the .NET Platform version on your web hosting or contact your web hosting provider.

On IIS 7:

1. Edits your website Basic Settings.
2. In the Application Pool field, select the application pool with the .NET Framework version required by C1 CMS (see “System Requirements”).

3. If you do not have such a pool, create it.

On IIS 6:

1. Open the Properties of your website.
2. Open the ASP.NET tab.

Figure 8: Selecting an application pool with .NET Framework on IIS 7
3. In the ASP.NET version field, select “4.0.319” (or a similar value - please see “System Requirements”)

![Selecting the ASP.NET version for a C1 CMS website on IIS 6](image)

Figure 9: Selecting the ASP.NET version for a C1 CMS website on IIS 6

2.3 No Sufficient Permissions on the Website Directory

When you open your website, you can get the server error: “HTTP 500.19”, “Cannot read configuration file due to insufficient permissions”.
Figure 10: Error when having no sufficient permissions on the website directory

This error occurs if you do not have sufficient (modify, read, write etc) permissions on the website directory.

If you have access to IIS, do as described below. Otherwise, follow the standard procedure of setting permissions on website directories on your web hosting or contact your web hosting provider.

On IIS or in Windows explorer:

1. Edit permissions on the website directory.
2. Make sure that modify, execute, list, read, and write permissions are set for the identity your website is running under. (On IIS 6, it should be “NETWORK SERVICE” under which ASP.NET is running. On IIS 7, it should be the identity under which the application pool used by your site is running.)
2.4 No Write Permissions on the Website Directory

When you open your website, you can get the server error: “Access to the path 'C:\inetpub\C1-CMS\App_Data\Composite\Configuration\FirstTimeStart.xml' is denied.”, or something similar.

![Figure 11: Setting proper permissions on the website directory](image)
Figure 12: Error when having no write permissions on the website directory

This error occurs if you do not have write permissions on the website directory.

To troubleshoot it, do as suggested in “No sufficient permissions on the website directory”.

2.5 Not Run as an Application on IIS

When you open your website, you can get the server errors for example: “CS0246: The type or namespace name 'Composite' could not be found (are you missing a using directive or an assembly reference?)". 
Server Error in '/' Application.

Compilation Error

Description: An error occurred during the compilation of a resource required to service this request. Please review the following specific error details and modify your source code appropriately.

Compiler Error Message: CS0246: The type or namespace name 'Composite' could not be found (are you missing a using directive or an assembly reference?)

Source Error:

Line 1: <# Import Namespace="Composite" %>
Line 2: <# Import Namespace="Composite.Core.Routing" %>
Line 3: <# Import Namespace="Composite.Data" %>
Line 4: <# Import Namespace="Composite.Core.Configuration" %>
Line 5: <# Import Namespace="Composite.Data.Types" %>

Source File: c:\inetpub\Composite\default.aspx Line: 8

Show Detailed Compiler Output

Version Information: Microsoft .NET Framework Version:4.0.30319; ASP.NET Version:4.0.30319.1

Figure 13: Error when not running a C1 CMS website as an Application

This error occurs if run your C1 CMS website on IIS 7 as a Virtual Directory. This works on IIS 6, but on IIS 7 you should run a C1 CMS website in a subfolder as an Application.

If you have access to IIS, do as described below. Otherwise, make sure that a subfolder you run a C1 CMS website in is configured as an IIS Application and not a Virtual Directory by following the standard procedure of configuring your remote website on your web hosting or contact your web hosting provider.

On IIS 7, do one of the following:

1. Right-click the C1 CMS website’s subfolder.
2. In the context menu, click Convert to Application.
3. Fill out the fields and click OK.

or

1. Right-click the C1 CMS website you want to create a subfolder in.
2. In the context menu, click Add Application.
3. Fill out the fields and click OK.
Figure 14: Adding a C1 CMS website as an Application in a subfolder