

10Tec DocMounter 3.0

What's New

Contents

v3.0.0 2023-Jan-30 (Release).....	2
v3.0.12 2023-May-22 (Update 1)	6
Addendum: Tags used to classify changes	7

v3.0.0 | 2023-Jan-30 (Release)

The 4 most important changes in this major update of DocMounter are the followings:

- Compiling of help solutions directly to the MSHC format.
- Viewing of built MSHC helps with 10Tec Help Viewer.
- Support for high-resolution screens.
- Ability to create help files without assemblies (conceptual topics only).

These and all other changes are described in greater detail below.

1. [New] Now DocMounter builds help solutions directly to the files of the MSHC format. These help solutions can be viewed in 10Tec Help Viewer or Microsoft Help Viewer supplied with Visual Studio.

DocMounter no longer needs the external outdated Microsoft Help2 Compiler (HxComp.exe) with related utilities (such as HxReg.exe). The corresponding option used to specify the location of the HxReg tool was removed from the DocMounter options dialog.

The ability to specify DocSet attributes in the DocMounter project properties dialog was also removed because these attributes are not used in the MSHC format.

2. [New] To view built MSHC files, DocMounter launches 10Tec Help Viewer supplied with it.

You will find two new options related to configuring of a help viewer application used to view built help solutions in the DocMounter options dialog. They allow you to specify the full path to the help viewer application and its command line arguments. Note that in the general case you can use any application that supports displaying MSHC files. 10Tec Help Viewer supplied with DocMounter is just a default option.

DocMounter no longer uses Microsoft Document Explorer (DExplore.exe) to display built HxS help files. The corresponding option used to specify the location of DExplore.exe was removed from the DocMounter options dialog.

The ability to specify a Help Namespace in the DocMounter project properties dialog was also removed because it is no longer needed to view built MSHC files in 10Tec Help Viewer or Microsoft Help Viewer.

3. [New][Enhancement] Support for high-resolution screens was implemented. This includes enhanced control layout logic to properly process various screen resolutions and a special set of high-quality UI icons for screens with a pixel density of 192ppi or higher. The UI icon set was also updated to provide a more modern flat look of the whole app.

4. [New] This release of DocMounter allows you to create projects containing only conceptual topics. Thus, now you can use this tool to create MSHC files without .NET assemblies.

To create such a project, select the **New Project** command from the **File** menu and then click **Cancel** in the **Open Assemblies** dialog. DocMounter will open a message box asking you whether you want to create a project without assemblies. Answer **Yes** in this message box.

5. [New] The text topic editor based on the unsupported Quantum Whale Editor.NET component in the previous versions of DocMounter was upgraded to the latest version of this component which is now called SyntaxEdit and is provided by the [AlterNET Software](#) company. SyntaxEdit v8.1 incorporated into DocMounter provides many enhancements needed by this release of DocMounter, such as high DPI support.
6. [New] The **Options** dialog now allows you to select a font for the topic text editor. This feature is especially helpful if you want to choose the text editor font that looks best on a high-resolution screen.
7. [New] DocMounter 3.0 allows you to create a root page for your documentation. As a rule, root pages are used as an entry point in the documentation the user automatically sees when the help solution is

opened. The root page automatically becomes the page with the highest hierarchy in the Table of Contents in the built help solution. In other words, all top-level conceptual and namespace topics from a DocMounter project will be child pages of the root page.

To add a root page to your help solution, place an XHTML file with the name `root_page.htm` in the same directory in which the DocMounter project is located. Below is the template of a root page with some meta tags an MSHC root page must have:

```
<?xml version="1.0" encoding="utf-8"?>
<html>
<head>
  <title>ROOT PAGE TITLE</title>
  <meta name="Description" content="ROOT PAGE DESCRIPTION" />

  <meta name="Microsoft.Help.Locale" content="en-us" />
  <meta name="Microsoft.Help.TopicLocale" content="en-us" />
  <meta name="Microsoft.Help.SelfBranded" content="true" />

  <meta name="Microsoft.Help.TocParent" content="-1" />
  <meta name="Microsoft.Help.TocOrder" content="0" />
</head>
<body>

<!-- ROOT PAGE CONTENTS -->

</body>
</html>
```

8. [Enhancement] The 3D borders in some interface elements were replaced with the OS-styled borders where it was possible or with a 1-pixel flat border to provide a consistent modern flat look of the whole app in the latest versions of Windows.
9. [Enhancement] The topic text editor no longer checks spelling for texts enclosed into `<code>` tags because there is no big sense in doing this for code snippets. This improvement greatly reduces the number of squiggly underline elements that can distract attention a lot if a topic contains code blocks.
10. [Enhancement] If the saved position of the application window is outside of the current screen, the default window location is used instead of the saved one. This eliminates the problem when the DocMounter window is not visible on the desktop after changing the display, especially if the new display has a different pixel density.
11. [Enhancement] All DocMounter features are now implemented in one executable file – no more external DLLs implementing separate features like building a help solution or manual. The whole set of DocMounter files is the following now:
 - TenTec.DocMounter.exe – the main executable file.
 - TenTec.DocMounter.exe.config – the application configuration file.
 - TenTec.DocMounter.SpellChecker.dat – the spell checker vocabulary (contains correct words).
12. [Enhancement] The formatting of several styles in `Manual.css` was enhanced.
13. [Enhancement] If you right-click a project node to display the context menu for it, this node is selected automatically now.
14. [Enhancement] Topic pages became cleaner due to additional post-processing of page HTMLs generated by Sandcastle. Among the removed elements - the thin gradient rectangle at the bottom of a page header and the empty first column in tables describing enumeration members.

15. [Enhancement] The font of the **DocMounter Options** and **Project Properties** dialogs was changed to Tahoma 10pt because this is the common font used in all other application dialogs.
16. [Enhancement] The main menu was reorganized and supplemented with new commands for more logical structure. The **Build and View Help** command was moved from the **Run** menu to the **Build** menu so that the **Build** menu contains all commands related to building documentation. The **View Built Help** command is now in the **Tools** menu because it launches an external tool to view the built help file (10Tec Help Viewer). The **Tools** menu also contains new commands to open the corresponding folder in the Windows Explorer: **Open Manual Folder**, **Open Brief XML Documentation Folder**, **Open Full XML Documentation Folder**. The **Run** menu has been deleted as it did not contain any items after this menu reorganization.

In addition, some menu item shortcuts were changed, and new shortcuts were assigned. For example, now you can use F9 as the keyboard shortcut for the **View Built Help** command and F4 for the **Project Properties** dialog.

17. [Enhancement] When a new project is created, in most cases the developer imports member descriptions from XML comment files accompanying the documented libraries using the corresponding command from the **Project** menu. Now DocMounter suggests doing this operation automatically after choosing libraries for a new project.
18. [Enhancement] When you issue the **Build Manual** command, DocMounter first checks whether at least one document name for a manual is specified. If not, DocMounter informs about that and stops the build process.
19. [Enhancement] If you issued the **Build Manual** or **Build Brief/Full XML Documentation** command, DocMounter automatically opens the corresponding output folder with the generated files upon successful build.
20. [Enhancement] DocMounter no longer converts paths to output folders and project libraries to relative paths related to the project file location when the project is saved. This allows you to move the project file to any new location without losing references to project libraries and use unchangeable absolute paths to output folders.
21. [Enhancement] The previous versions of DocMounter may have created several temporary output folders while building help. If the option to delete all temporary files after successful build was not set, you could get these folders in different places. Now all temporary files are created only in one folder named 'build.temp' inside the output folder related to a particular build process.
22. [Fixed] The Media folder in the manual output folder is created only if required.
23. [Fixed] Various exceptions may have been thrown if the project file could not be loaded correctly (because of errors in the project file itself, when the referenced libraries could not be loaded, etc.)
24. [Fixed] If DocMounter found a problem during loading a project file and asked to cancel project loading operation, it continued to load the project even if the operation was cancelled.
25. [Fixed] The previous versions of DocMounter may have not imported assembly member descriptions from accompanying XML comments files for the following members:
 - class constructors and protected methods;
 - generic type arguments of methods;
 - arguments of delegates.

This release of DocMounter fixes all revealed problems with XML comments import.

26. [Fixed] The project node tree flickered in Windows 11 while the mouse pointer moved over its nodes.

27. [Fixed] The **Insert Tag** toolbar button and menu item were not disabled correctly when text editor fields lost the input focus.
28. [Fixed] The **Insert Tag** dialog inserts the correct language code for Visual Basic code snippets now.

v3.0.12 | 2023-May-22 (Update 1)

1. [New] The MSHC build log displayed in the **Output** pane now contains the build time and the total number of warnings from Sandcastle.
2. [New] The **Edit** menu contains the new **Make Bold** command with the **Ctrl+B** keyboard shortcut. This command can be used to quickly insert the bold text tag () into the text editor or wrap the selected text with it.
3. [Enhancement] If the conceptual topic tree in the **Insert Tag** dialog has the only root node, it is expanded automatically to save one user click and make selection of a conceptual topic more comfortable.
4. [Fixed] If you worked with a project without assemblies, exceptions were thrown when you opened the **Insert Tag** dialog or issued the commands to build XML documentation.
5. [Fixed] The namespace tree in the main window and in the **Member Topics** section of the **Insert Tag** dialog were not updated after changing the library list in the **Project Properties** dialog.
6. [Fixed] The **Insert Tag** dialog may not have been available after repopulating the project tree in the main window.
7. [Fixed] DocMounter tags containing links to pages with method overload lists in the Microsoft documentation (for example, <see cref="Overload:System.Convert.ToDecimal" />) were not processed correctly.
8. [Fixed] MSHC links to enumeration fields in the Microsoft online documentation now work: they open the corresponding enumeration description page (because the current version of the Microsoft online documentation do not provide separate pages for enumeration fields).
9. [Fixed] If you opened the **Conceptual Topic Properties** dialog and clicked **OK** in this dialog, the changes in the topic editor were abandoned.
10. [Fixed] Links to Microsoft Docs for generic members were not created correctly.
11. [Fixed] Pressing F2 opened the **Conceptual Topic Properties** dialog even if another dialog (such as **Insert Tag**) was opened.
12. [Fixed] Namespaces from <see cref=""> tags were not inserted "as is" into manuals. They were reduced using the rules for methods and properties, when only the part after the last dot was inserted.

Addendum: Tags used to classify changes

- [New] - Completely new feature.
- [Change] - Change in a member functionality or interactive behavior.
- [Fixed] - Fixed bug or solved problem.
- [Removed] - Member was completely removed.
- [Enhancement] - Some functionality was enhanced.
- [Optimization] - Performance was improved.
- [Renaming] - Member was renamed.
- [Code-Upgrade] - Source code for previous versions may require changes.