Upgrading WebWorks ePublisher

Maintaining Your Overrides

Revision B

toojaded.com

toojaded.com

Upgrading WebWorks ePublisher:Maintaining Your Overrides

Rev. B

Revised: 28 September 2012

Contents

Upgrading WebWorks ePublisher

Author: Nadine Murray

So you upgraded to the next version of WebWorks ePublisher either because you received the next version in your subscription, or you are upgrading for the first time in a few years. Then you open an existing project and you get a message.

```
This project contains custom code. To ensure compatibility
with ePublisher 2012.2, the project is now configured to use
formats from ePublisher 2011.4.

The project format version can be reconfigured via the Project
Settings dialog.
```

You scratch your head and think, “What? Oh, I can roll it forward by going to Projects > Project Settings, and then change the value to 2012.2 and everything will be all right.”

Well, no. What this message means is that you have overrides or customizations in your project that could be susceptible to code conflicts when WebWorks updates ePublisher. You need to do some housekeeping while you develop your projects and whenever you upgrade to a newer version of ePublisher by maintaining any files that you customize. A little effort when you customize files in your project and when you upgrade ePublisher can save you a lot of trouble in the long run. This housekeeping consists of four steps.

1 Creating overrides

2 Comparing your overrides to the installation files

3 Mitigating the differences between your overrides and the installation files

4 Regenerating your project

REMEMBER: If you do not have time to maintain your overrides when you upgrade ePublisher, you can hold your project at the previous release version until you do have time. This is also known as “compatibility mode”. However, the sooner you upgrade to the new version, the sooner you can take advantage of the new features that have been added to ePublisher.

Terminology

base files

ePublisher files that are in the \\Program Files\WebWorks folder. Also called installation files. Do not change these files.

compatibility mode

When you run your project at a version that is earlier than the version of ePublisher that is installed on your computer, you are running it in compatibility mode. You can change the version at which you run your project by going to **Project > Project Settings > General tab > Compatibility Configuration**.

customizations

Changes that you make to the underlying ePublisher files in your project in the Targets and/or Formats folder. Also called overrides.

installation files

ePublisher files that are in the \\Program Files\WebWorks folder. Also called base files. Do not change these files.

overrides

Changes that you make to the underlying ePublisher files in your project in the Targets and/or Formats folder. Also called customizations.

super customization

Minimal override that isolates custom XSL code, and that allows you to override an XSL file without mixing custom code with installation code. Also called wwtransform:super.

wwtransform:super

Extension to ePublisher URI Resolvers that extends the tiered look-up behavior to allow you to import the lower-priority version of the file in addition to specializing some aspect of it in a higher version. Also called super customization.

Out-of-the-Box Test

After I have installed the newest version of ePublisher, but before I maintain any overrides, I like to run an out-of-the-box test. To do this, I create a fresh project in ePublisher that has no changes or customizations at all (or as few as possible). I add a representative sample of source documents to the project. I also examine the release notes for information about new functionality or fixes that might make my overrides redundant.

When I examine this default output, it often gives me an idea of what overrides I can expect to keep and what I can expect to delete from my project. It also gives me feedback about other adjustments that I might need to make to my project either through the user interface, or by creating new overrides.

Managing Overrides

Up until ePublisher 2011.4, users had to manage overrides manually. In ePublisher 2012.1, WebWorks added an Advanced menu that lets users manage overrides through the user interface. In ePublisher 2012.2, WebWorks added more options to the Advanced menu. Wherever it is appropriate, I describe managing overrides manually and through the 2012.2 options of the Advanced menu.

Advanced Menu

NOTE: The Advanced menu was introduced in ePublisher 2012.1. More options were added in ePublisher 2012.2. This topic describes the options in ePublisher 2012.2.

The Advanced menu lets you manage your overrides through the ePublisher interface. Most of the items on the Advanced menu are available on the menu that appears when you right-click a file in the **Manage Target Customizations** or **Manage Format Customizations** tab.

Advanced Menu Items

Item	Description
Manage Format Customizations	Opens a tab that displays the files that are available for customization for the output format that is currently selected. Any files that contain overrides appear in bold font. Use this tab to: • select the files that you want to customize. • edit override files. • compare override files to the installation files. • remove override files from your project. If the folders and file appear blue, it means that the override file is located within the folder of the specified format, and that either: • the override file is unique to your project and does not have an equivalent installation file. • the file exists in both the installation folder and your override folder, but the file path for each is different. If an override file is located in the Formats folder, but is not within the folder of the specified format, then it does not appear on the Manage Format Customizations tab. For more information, see “Types of Overrides” on page 5.

Item	Description
Manage Target Customizations	Opens a tab that displays the files that are available for customization for the target that is currently selected. Any files that contain overrides appear in bold font. Use this tab to: • select the files that you want to customize. • edit override files. • compare override files to the installation files. • remove override files from your project. If the folders and file appear blue, it means that the override file is located within the folder of the specified target, and that either: • the override file is unique to your project and does not have an equivalent installation file. • the file exists in both the installation folder and your override folder, but the file path for each is different. If an override file is located in the Targets folder, but is not within the folder of the specified target, then it does not appear on the Manage Target Customizations tab. For more information, see "Types of Overrides" on page 5.
Explore To Customizations	Open the folder in Windows Explorer where the selected file is located.
Edit File	Opens the file in the editor that you specified in the WebWorks ePublisher Preferences dialog box so that you can make changes to the file. In addition, you can open the file by double-clicking the file name. **IMPORTANT:** If you open the file, but do not edit it, ePublisher still considers the file to be customized and displays the file in bold font.
View File Differences	Opens the selected file and the equivalent installation file in the file comparison utility that you specified in the WebWorks ePublisher Preferences dialog box.
Create Customization	Makes a copy of the selected file in either the Targets or Formats folder in your project so that you can customize the file. The file path of the installation file is mirrored in the process. If the Targets or Formats folder does not exist in your project, it is created automatically. This process only copies the file into your project; it does not open the file for editing. The file appears in bold font in the **Manage Target Customizations** or **Manage Format Customizations** tab.

Item	Description
Create Super Customization	Creates a skeletal transform that is configured to let you make super customizations using the wwtransform:super mechanism. The file path of the installation file is mirrored in the process. If the Targets or Formats folder does not exist in your project, it is created automatically. This process only creates the file in your project; it does not open the file for editing. The file name appears in bold font in the **Manage Target Customizations** or **Manage Format Customizations** tab.
Remove Customization	Deletes the file from your override folder. ePublisher now uses the file that is in the installation folder when you generate output.
Configure Tools	Opens the WebWorks ePublisher Preferences dialog box. The **Editor Preferences** tab lets you specify which text editor to use for editing override files, and which image editor to use for opening images. By default, ePublisher opens the application that is associated with that file or image's file type in Windows. The **Diff Preferences** tab lets you specify which file comparison tool to use when you compare your override files to the installation files. In addition, you can open the WebWorks ePublisher Preferences dialog box by going to **Edit > Preferences**.

Types of Overrides

You can create overrides that affect your project in three ways depending on where you place your override files: project wide, per format, and per target. Where you place your overrides affects whether or not they appear in the **Manage Format Customizations** and **Manage Target Customizations** tabs, and how they appear.

Types of Overrides

Override Type	Override Placement	Visibility	Example
Project wide	`\\Formats\[override]`	Not visible	`\\Formats\Adapters`
Format wide	`\\Formats\[output format]\[override]`	Bold or blue	`\\Formats\Reverb\Adapters`
Target wide	`\\Targets\[target]\[override]`	Bold or blue	`\\Targets\ProductA\Pages\Page.asp`

Appearance of Overrides in the Manage Customization Tabs

Appearance	Meaning
Bold	The override has an equivalent installation file and the file path is mirrored in the target or format override folder.
Blue	• The override has an equivalent installation file, but the file path is not mirrored in the target or format override folder. For example, the Adapters or Shared transforms folders. • The override file does not have an equivalent installation file, but is located in the target or format folder. You cannot add these override files to your project using the Advanced menu; you must add them manually. If you want to view the file differences using the Advanced menu, you must locate the installation file manually from within your file comparison utility.
Not visible	The override file is located at the top level of the Formats folder, not within the folder of the specified format. For example, `\\Formats\Adapters`. You cannot add or manage these types of overrides using the Advanced menu; you must add and manage them manually.

“Blue” Override Example

If you create an override to a file in the installation’s Shared folder, and then manually copy the Shared folder into your target or format folder, the file path to the override file does not mirror the file path to the equivalent installation file. An example of this would be any customizations that you make to the locales.xml file.

Override file path:

```
C:\pdf_project\Formats\PDF - XSL-FO\Shared\common\locale\locales.xml
```

Installation file path:

```
C:\Program Files
\WebWorks\ePublisher\2012.2\Formats\Shared\common\locale\locales.xml
```

The folders and file name appear blue in the **Manage Target Customizations** or **Manage Format Customizations** tab. If you want to view the file differences, you must locate the installation file manually from within your file comparison utility.

Adapter Override Example

For an example of when you might use the Adapter folder in a project, see Mapping DITA Classes to ePublisher Styles in the WebWorks online help.

Creating Overrides

Sometimes it is necessary to create overrides (or customizations) in underlying ePublisher files. If you create overrides using ePublisher 2011.4 or earlier, make sure that you override **copies** of the installation files in a Targets override folder and/or Formats override folder in your project.

IMPORTANT: DO NOT customize the installation files in \\Program Files\WebWorks.

Targets Folder vs. Formats Folder

- Overrides in the Targets folder apply to the specified target output, such as ProductA versus ProductB.
- Overrides in the Formats folder apply to the type of output, such as WebWorks Help 5.0 or Reverb output.
- You can have both a Targets folder and a Formats folder in your project.
- If they conflict, then overrides in the Targets folder take precedence over overrides in the Formats folder.

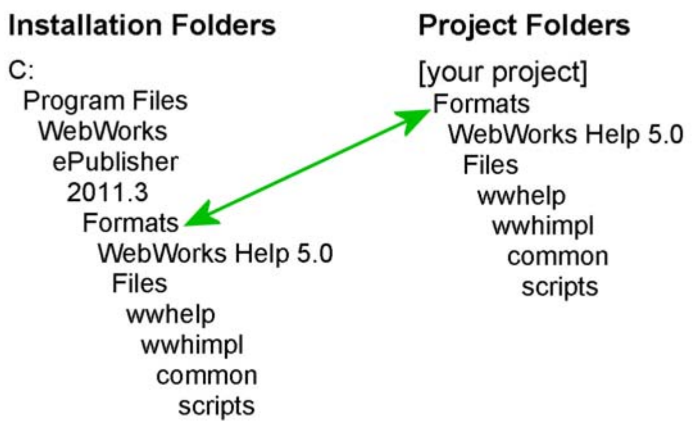
For information about creating override folders manually, go to Creating Project Target Overrides on www.webworks.com.

For information about creating override folders using the Advanced menu, go to Project Format and Target Overrides in the WebWorks online help.

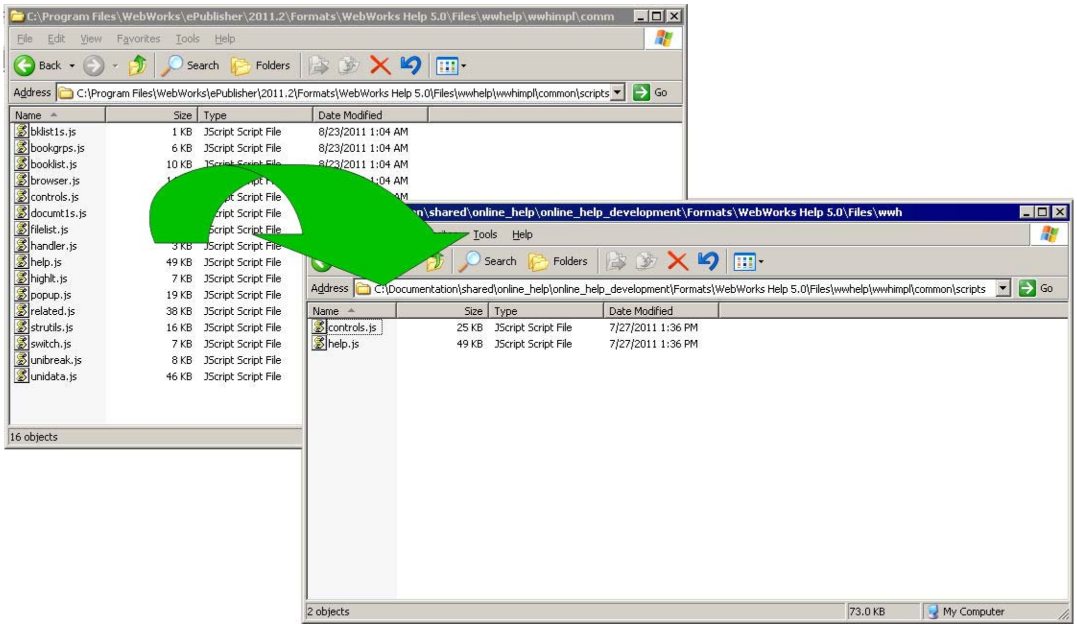
Take Only What You Need

When you create overrides for your project *manually*, copy only the files that you need to customize. For example, if you need to customize the **control.js** and **help.js** files, copy only those files; do not copy all of the files in the same folder.

Regardless of whether you are creating a Targets or Formats folder in your project, mirror the folder hierarchy of the format that is in the installation folder.



Take only the folders and files that you need to override



C:\Program Files\WebWorks\ePublisher\2011.2\Formats\WebWorks Help 5.0\Files\wwhelp\wwhimpl\comm

Address: C:\Program Files\WebWorks\ePublisher\2011.2\Formats\WebWorks Help 5.0\Files\wwhelp\wwhimpl\common\scripts

Name	Size	Type	Date Modified
bklist1s.js	1 KB	JScript Script File	8/23/2011 1:04 AM
bookgrps.js	6 KB	JScript Script File	8/23/2011 1:04 AM
booklist.js	10 KB	[illegible]	8/23/2011 1:04 AM
browser.js	[illegible]	[illegible]	[illegible]
controls.js	[illegible]	[illegible]	[illegible]
documt1s.js	[illegible]	[illegible]	[illegible]
filelist.js	[illegible]	[illegible]	[illegible]
handler.js	3 KB	JScript Script File	[illegible]
help.js	49 KB	JScript Script File	[illegible]
highlt.js	7 KB	JScript Script File	[illegible]
popup.js	19 KB	JScript Script File	[illegible]
related.js	38 KB	JScript Script File	[illegible]
strutils.js	16 KB	JScript Script File	[illegible]
switch.js	7 KB	JScript Script File	[illegible]
unibreak.js	8 KB	JScript Script File	[illegible]
unidata.js	46 KB	JScript Script File	[illegible]

16 objects

...n\shared\online_help\online_help_development\Formats\WebWorks Help 5.0\Files\wwh

Address: C:\Documentation\shared\online_help\online_help_development\Formats\WebWorks Help 5.0\Files\wwhelp\wwhimpl\common\scripts

Name	Size	Type	Date Modified
controls.js	25 KB	JScript Script File	7/27/2011 1:36 PM
help.js	49 KB	JScript Script File	7/27/2011 1:36 PM

2 objects 73.0 KB My Computer

Comment Your Code

When you customize code, make sure that you comment the custom code meticulously.

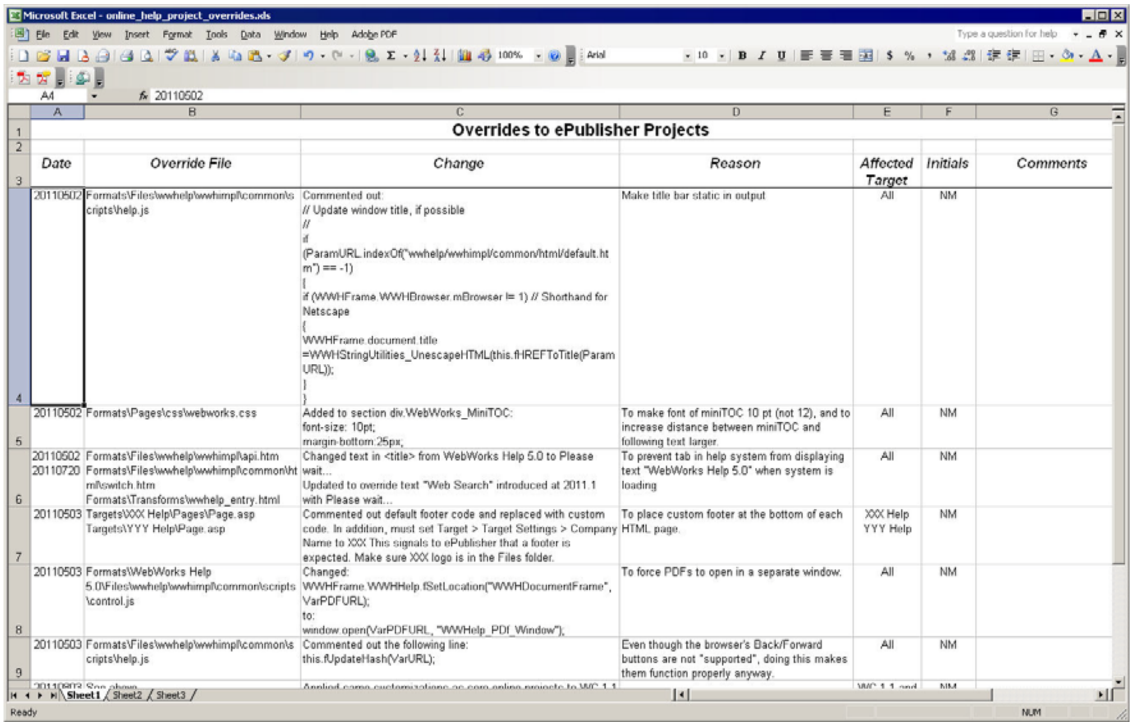
```
VarPDFURL = VarDocumentParentURL + "/" + VarPDFLink;
// 20110503 NM changed following line to force PDFs to open in a separate
window //
window.open(VarPDFURL, "WWHelp_PDf_Window");
```

Keep a record in another file, such as a txt file, of the files that you changed with a brief description of changes you made. Store this file along with your project, and print it out the next time you upgrade ePublisher to help you remember what files you changed and what changes you made.

Detailed Descriptions of Customizations

I like to create a descriptive list of what I changed in a spreadsheet. I include information about why overrides were created, what targets are affected, where I found information about the override, and any other comments about the override that I might need to know in the future.

REMEMBER: A detailed record should not replace commenting your code changes. Always comment your code changes within your override files.


Microsoft Excel - online_help_project_overrides.xls
File Edit View Insert Format Tools Data Window Help Adobe PDF
Type a question for help
100%
Arial
10
A4
20110502
Sheet1
Sheet2
Sheet3
Ready
NUM

Overrides to ePublisher Projects

Date	Override File	Change	Reason	Affected Target	Initials	Comments

20110502	Formats\Files\wwhelp\wwhimpl\common\scripts\help.js	Commented out:
		// Update window title, if possible
		//
		if
		(ParamURL.indexOf("wwhelp/wwhimpl/common/html/default.htm") == -1)
		{
		if (WWHFrame.WWHBrowser.mBrowser != 1) // Shorthand for Netscape
		{
		WWHFrame.document.title
		=WWHStringUtilities_UnescapeHTML(this.fHREFToTitle(ParamURL));
		}
		}
			Make title bar static in output	All	NM	

20110502	Formats\Pages\css\webworks.css	Added to section div.WebWorks_MiniTOC:
		font-size: 10pt;
		margin-bottom:25px;
			To make font of miniTOC 10 pt (not 12), and to increase distance between miniTOC and following text larger.	All	NM	

20110502 20110720	Formats\Files\wwhelp\wwhimpl\api.htm Formats\Files\wwhelp\wwhimpl\common\html\switch.htm Formats\Transforms\wwhelp_entry.html	Changed text in <title> from WebWorks Help 5.0 to Please wait... Updated to override text "Web Search" introduced at 2011.1 with Please wait...	To prevent tab in help system from displaying text "WebWorks Help 5.0" when system is loading	All	NM	
20110503	Targets\XXX Help\Pages\Page.asp Targets\YYY Help\Page.asp	Commented out default footer code and replaced with custom code. In addition, must set Target > Target Settings > Company Name to XXX This signals to ePublisher that a footer is expected. Make sure XXX logo is in the Files folder.	To place custom footer at the bottom of each HTML page.	XXX Help YYY Help	NM	

20110503	Formats\WebWorks Help 5.0\Files\wwhelp\wwhimpl\common\scripts\control.js	Changed:
		WWHFrame.WWHHelp.fSetLocation("WWHDocumentFrame", VarPDFURL);
		to:
		window.open(VarPDFURL, "WWHelp_PDf_Window");
			To force PDFs to open in a separate window.	All	NM	

20110503	Formats\Files\wwhelp\wwhimpl\common\scripts\help.js	Commented out the following line: this.fUpdateHash(VarURL);	Even though the browser's Back/Forward buttons are not "supported", doing this makes them function properly anyway.	All	NM	

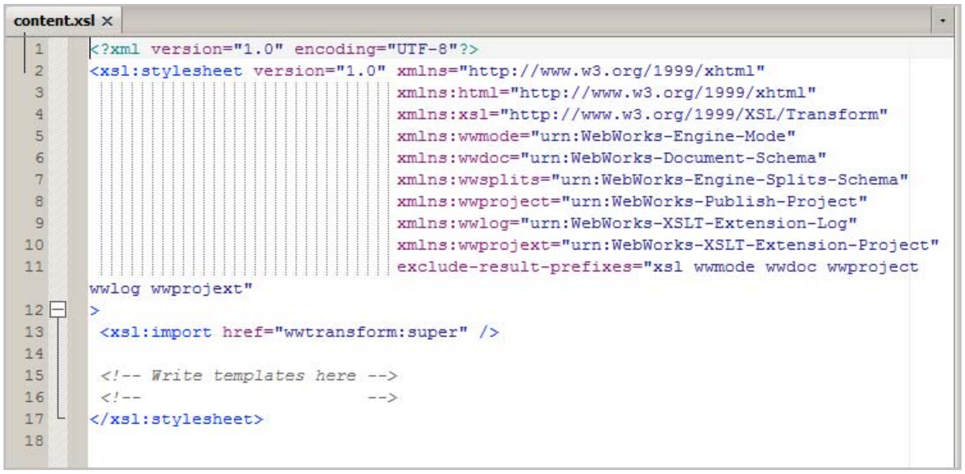
It might not always be possible to maintain such a detailed list of your overrides because it takes time to complete; however, knowing what has changed, where, and why will be invaluable to you when you upgrade your version of ePublisher, and to other people when they take over your project.

Super Customizations

Super customizations (also referred to as wwtransform:super) are minimal overrides that let you isolate custom XSL code. The advantage of super customizations is that it lets you override an XSL file without mixing your custom code with the code in the installation file. This makes such customizations easier to maintain.

You can create a super customization file manually, or by using the Advanced menu if your use ePublisher 2012.1 or later. If you use the Advanced menu, ePublisher creates a skeletal transform that is configured to let you create customizations using the wwtransform:super mechanism.

Example super customization skeletal transform of content.xsl file



content.xsl ×

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns="http://www.w3.org/1999/xhtml"
                              xmlns:html="http://www.w3.org/1999/xhtml"
                              xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
                              xmlns:wwmode="urn:WebWorks-Engine-Mode"
                              xmlns:wwdoc="urn:WebWorks-Document-Schema"
                              xmlns:wwsplits="urn:WebWorks-Engine-Splits-Schema"
                              xmlns:wwproject="urn:WebWorks-Publish-Project"
                              xmlns:wwlog="urn:WebWorks-XSLT-Extension-Log"
                              xmlns:wwprojext="urn:WebWorks-XSLT-Extension-Project"
                              exclude-result-prefixes="xsl wwmode wwdoc wwproject
wwlog wwprojext"
>
 <xsl:import href="wwtransform:super" />

 <!-- Write templates here -->
 <!--                      -->
</xsl:stylesheet>
```

The code in the super customization file takes precedence over code in the installation file. The installation file's code is invoked only if you use <xsl:apply-imports>. This allows you to process the code by using the templates in the underlying stylesheet.

Super customizations can be applied only to XSL files.

For more information about super customizations, go to the WebWorks wiki article called wwtransform:super.

These WebWorks wiki articles describe use cases that use super customizations.

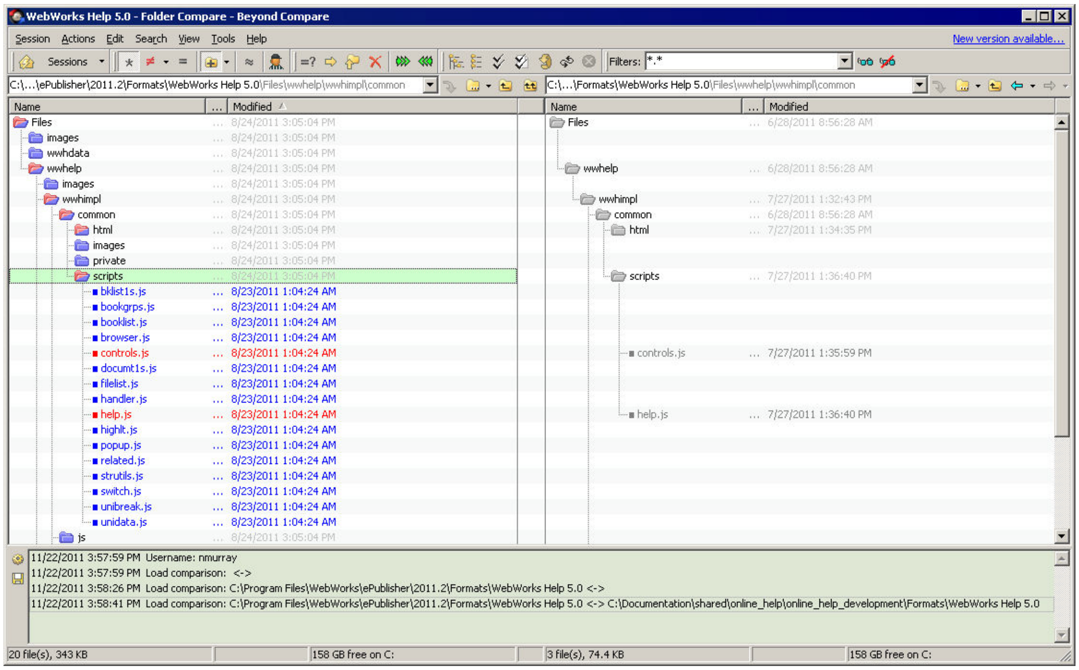
- 2-Celled Note Table
- Custom Marker Processing

Comparing Your Overrides to the Installation Files

NOTE: I strongly recommend that you back up your override folders to another location on your computer before you start comparing overrides to the new installation files. If you want to roll back changes that you make, you can restore your previous overrides from the files that you backed up.

When you upgrade your version of ePublisher, you must take some time to compare your customized folders and files to the corresponding folders and files in the WebWorks installation folder. Use a file comparison utility to do this. I use Beyond Compare, but there are many other tools available for this task (http://en.wikipedia.org/wiki/Comparison_of_file_comparison_tools).

Compare the WebWorks installation folders and files to your override folders and files


WebWorks Help 5.0 - Folder Compare - Beyond Compare
Session Actions Edit Search View Tools Help
New version available...
Sessions
Filters: *.*

C:\...\ePublisher\2011.2\Formats\WebWorks Help 5.0\Files\wwhelp\wwhimpl\common

C:\...\Formats\WebWorks Help 5.0\Files\wwhelp\wwhimpl\common

Name	...	Modified	Name	...	Modified
Files	...	8/24/2011 3:05:04 PM	Files	...	6/28/2011 8:56:28 AM
images	...	8/24/2011 3:05:04 PM			
wwhdata	...	8/24/2011 3:05:04 PM			
wwhelp	...	8/24/2011 3:05:04 PM	wwhelp	...	6/28/2011 8:56:28 AM
images	...	8/24/2011 3:05:04 PM			
wwhimpl	...	8/24/2011 3:05:04 PM	wwhimpl	...	7/27/2011 1:32:43 PM
common	...	8/24/2011 3:05:04 PM	common	...	6/28/2011 8:56:28 AM
html	...	8/24/2011 3:05:04 PM	html	...	7/27/2011 1:34:35 PM
images	...	8/24/2011 3:05:04 PM			
private	...	8/24/2011 3:05:04 PM			
scripts	...	8/24/2011 3:05:04 PM	scripts	...	7/27/2011 1:36:40 PM
bklist1s.js	...	8/23/2011 1:04:24 AM			
bookgrps.js	...	8/23/2011 1:04:24 AM			
booklist.js	...	8/23/2011 1:04:24 AM			
browser.js	...	8/23/2011 1:04:24 AM			
controls.js	...	8/23/2011 1:04:24 AM	controls.js	...	7/27/2011 1:35:59 PM
documt1s.js	...	8/23/2011 1:04:24 AM			
filelist.js	...	8/23/2011 1:04:24 AM			
handler.js	...	8/23/2011 1:04:24 AM			
help.js	...	8/23/2011 1:04:24 AM	help.js	...	7/27/2011 1:36:40 PM
highlt.js	...	8/23/2011 1:04:24 AM			
popup.js	...	8/23/2011 1:04:24 AM			
related.js	...	8/23/2011 1:04:24 AM			
strutils.js	...	8/23/2011 1:04:24 AM			
switch.js	...	8/23/2011 1:04:24 AM			
unibreak.js	...	8/23/2011 1:04:24 AM			
unidata.js	...	8/23/2011 1:04:24 AM			
js	...	8/24/2011 3:05:04 PM			

```
11/22/2011 3:57:59 PM  Username: nmurray
11/22/2011 3:57:59 PM  Load comparison:  <->
11/22/2011 3:58:26 PM  Load comparison: C:\Program Files\WebWorks\ePublisher\2011.2\Formats\WebWorks Help 5.0 <->
11/22/2011 3:58:41 PM  Load comparison: C:\Program Files\WebWorks\ePublisher\2011.2\Formats\WebWorks Help 5.0 <-> C:\Documentation\shared\online_help\online_help_development\Formats\WebWorks Help 5.0
```

20 file(s), 343 KB | 158 GB free on C: | 3 file(s), 74.4 KB | 158 GB free on C:

Comparing Folders Manually

If your are using ePublisher 2011.4 or earlier, you must compare folders and files manually.

1. Minimize the differences against the WebWorks installation files by comparing the folders in the WebWorks installation folders to your override folders.
2. Delete any files and folders in your override folders that do not actually contain any customizations.
3. Identify any files in your override folders that must be inspected for code conflicts.

Comparing Folders Using the Advanced Menu

You must set up ePublisher to open your file comparison utility before you can compare your overrides to the installation files using the Advanced menu. You can do this on the **Diff Preferences** tab of the WebWorks ePublisher Preferences dialog box.

1. You must “teach” ePublisher which installation folder to look in when you compare your overrides.

   a. Go to **Projects > Project Settings**.

      The Project Settings dialog box appears.

   b. In the **Base format version** field on the **General** tab, select the most current version of ePublisher.

   c. Save your project, and then close it.

d. Reopen your project.

Now when you compare your override folder and files, the correct installation folder opens in your file comparison utility.

TIP: If you decide not to perform this step, your file comparison utility opens to the installation folder of the ePublisher version that is set in the Project Settings dialog box. If it is not set to the most current version, you must manually locate the most current version folder from within your file comparison utility.

2. Go to **Advanced**.

- To see which files contain overrides for the selected target, choose **Manage Target Customizations**.
- To see which files contain overrides for the selected format, choose **Manage Format Customizations**.

A tab appears that displays the files available for customization. The files that contain overrides appear in bold font.

Diff Reports

Many comparison utilities let you generate a report that indicates which folders are different between the installation folder and your project folder. In addition, you can generate a report of the differences between the files that you are comparing. You can save these reports, usually as plain text or HTML to review later so you do not have to rerun the comparison.

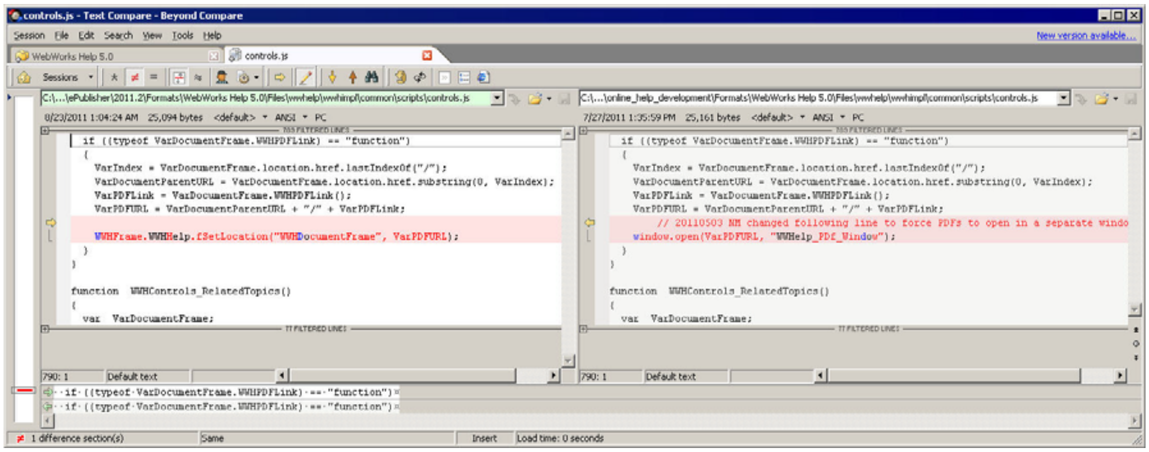
Mitigating the Differences Between Your Overrides and the Installation Files

Now that you have identified which files contain overrides, you must compare the code in the new installation files with the code in your customized files. This task can be daunting if you are not familiar with what the new code is doing. However, if you invest a bit of time now, you will save yourself a lot of trouble when a deadline is looming.

Use your file comparison tool to compare your customized files to the corresponding installation files in the newest version of the WebWorks folder.

IMPORTANT: Perform this step with great care. You do not want to delete or overwrite your customizations by accident; you only want to delete or overwrite them on purpose.

Compare the code of your override to the new code in the corresponding WebWorks file


controls.js - Text Compare - Beyond Compare
Session File Edit Search View Tools Help
New version available...
WebWorks Help 5.0
controls.js
Sessions
C:\...\ePublisher\2011.2\Formats\WebWorks Help 5.0\Files\wwhelp\wwhimpl\common\scripts\controls.js
8/23/2011 1:04:24 AM 25,094 bytes <default> ANSI PC
C:\...\online_help_development\Formats\WebWorks Help 5.0\Files\wwhelp\wwhimpl\common\scripts\controls.js
7/27/2011 1:35:59 PM 25,161 bytes <default> ANSI PC
790: 1 Default text
1 difference section(s) Same Insert Load time: 0 seconds

```
  if ((typeof VarDocumentFrame.WWHPDFLink) == "function")
  {
    VarIndex = VarDocumentFrame.location.href.lastIndexOf("/");
    VarDocumentParentURL = VarDocumentFrame.location.href.substring(0, VarIndex);
    VarPDFLink = VarDocumentFrame.WWHPDFLink();
    VarPDFURL = VarDocumentParentURL + "/" + VarPDFLink;

    WWHFrame.WWHHelp.fSetLocation("WWHDocumentFrame", VarPDFURL);
  }
}

function  WWHControls_RelatedTopics()
{
  var  VarDocumentFrame;
```

```
  if ((typeof VarDocumentFrame.WWHPDFLink) == "function")
  {
    VarIndex = VarDocumentFrame.location.href.lastIndexOf("/");
    VarDocumentParentURL = VarDocumentFrame.location.href.substring(0, VarIndex);
    VarPDFLink = VarDocumentFrame.WWHPDFLink();
    VarPDFURL = VarDocumentParentURL + "/" + VarPDFLink;
        // 20110503 NM changed following line to force PDFs to open in a separate windo
    window.open(VarPDFURL, "WWHelp_PDf_Window");
  }
}

function  WWHControls_RelatedTopics()
{
  var  VarDocumentFrame;
```

```
··if·((typeof·VarDocumentFrame.WWHPDFLink)·==·"function")¤
··if·((typeof·VarDocumentFrame.WWHPDFLink)·==·"function")¤
```

1. Back up your override files to a different location on your computer.
2. Compare your override files to the installation files.
   - If you are comparing files manually, open your override file and the corresponding file in the installation folder from within your file comparison utility.
   - If you are using ePublisher 2012.2 or later, use the **View File Differences** item on the Advanced menu.
3. Decide whether or not to accept the new code.
   - If WebWorks has changed the code so that it does the same thing as your override, overwrite your custom code with WebWorks’ new code.
   - If WebWorks has not changed the code and you still need your override, keep your override code.
   - If WebWorks has changed the code or added new code to the file in general, and you cannot tell what it does, then go to “What the Heck is That Code Doing?” on page 14 or “Three-way Merge” on page 14.

Regenerating Your Project

Once you have mitigated the differences between your override files and the new WebWorks installation files, regenerate your project in ePublisher. If your output does not function as you expect it to, or if no output is created at all, you must go back and compare your customized files to the new WebWorks installation files again.

On the other hand, if everything works to your satisfaction, you are done and you can now update your stationery.

What the Heck is That Code Doing?

If you are like me, you find code pretty baffling. Quite often, when WebWorks introduces new code, I have no idea what it is doing. To get around this, I break down my process. It takes a bit of time, but it is worth it in the end.

NOTE: Make sure that you back up your override files to another location on your computer.

1. Mitigate the code that you understand.
2. Make sure that you can still generate your output in ePublisher as expected.
3. Back up your override files again.
4. Return to the files that have new code.
5. Add the new code to your override file, and then regenerate your project in ePublisher.
6. If the project does not create output as expected, copy your backup files back into your override folders so that you can start adding the new code all over again.
7. Add the new code in increments.

   For example, one file at a time. Or if there is a defined section within the file, make the changes in that section.
8. Regenerate your project in ePublisher.
9. Repeat steps 6 to 8 until all of the new code is added to your override files and your output is generated as expected.

CAUTION: This approach usually works for me. However, code in one override file might depend on code in another override file. In which case, adding new code incrementally might prevent your output from generating properly.

Three-way Merge

The WebWorks installation folder provides you with a reference library of legacy formats.

One benefit of this is that it allows you to compare one version of an installation file to another version of the same installation file to see if WebWorks made any changes to the code. For example, you can compare two versions of the Page.asp file.

```
C:\Program Files\WebWorks\ePublisher\2011.2\Formats\WebWorks Help
5.0\Pages\Page.asp
```

```
C:\Program Files\WebWorks\ePublisher\2012.2\Formats\WebWorks Help
5.0\Pages\Page.asp
```

If there are no changes between installation versions, then you know you do not have to update your customized file.

Another benefit of the reference library is that it allows you to use your comparison utility to perform a three-way merge, if your utility supports that function. The three-way merge can help you maintain your overrides even if you do not fully understand the changes to the code.

For more information about three-way merge, see Lauren Lever's blog Wrangling the 3-way merge.

More Information

For a demonstration of maintaining your custom code, go to Maintenance of ePublisher code overrides Power Hour recording on www.webworks.com. Even though it is a little dated, this demonstration provides a good description of how to maintain overrides.

Acknowledgment

Thank you to Ben Allums of WebWorks for providing the information for this article, reviews, arguments, and encouragement. *C'est vraiment gentil de ta part*.

Thank you also to Jesse Wiles of WebWorks for setting me straight on super customizations.

Revision History

Revision	Date	ePublisher Version	Description
A	March 2012	2011.4	• Initial draft.
B	September 2012	2012.2	• Added content about Advanced menu. • Noted differences between managing overrides manually and using the Advanced menu. • Added content about Super Customizations.