2023

Upgrading your WebWorks ePublisher project

MAINTAINING YOUR OVERRIDES

NADINE MURRAY

Revision D

Contents

Upgrading your WebWorks ePublisher project

When you open an existing Designer project in a new 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, you might see the following message.

```
This project is configured to use Legacy Format version 2021.1. This
setting can be modified in Project Settings.
```

You might think, “What? Oh, I can roll it forward by going to **Projects > Project Settings**, change the value to the newest version of Designer, and then everything will be all right.”

Well, no. What this message means is that you have overrides (or customizations) in your project that could cause code conflicts each time that WebWorks updates ePublisher.

To prevent code conflicts with any overrides that you created, you need to do some housekeeping while you develop your project. And then each time that WebWorks upgrades ePublisher, you need to maintain the files that you customized in the previous version. A little effort when you create overrides in your project and when you upgrade ePublisher can save you a lot of trouble in the long run.

To maintain your overrides, you do the following:

1. Compare your overrides to the installation files.
2. Reduce the differences between your overrides and the installation files.
3. Regenerate your project.

> **TIP:** 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 that you upgrade to the new version, the sooner you can take advantage of the new features that have been added to ePublisher.

Watch this video before you begin

WebWorks has created a very useful video called *Advanced Customizations for WebWorks*. This video describes how to reduce code difference between overrides (or customizations) in your project and the new ePublisher installation files. I encourage you to watch this video before you read this article. The WebWorks video in conjunction with the instructions in this article will help you to maintain your projects every time you upgrade your version of ePublisher.

Advanced Customizations for WebWorks Reverb 2 (Long)

Terminology

base files

ePublisher files that are located in the `\\Program Files\WebWorks` folder. In this article, the base files are called *installation files*.

CAUTION: Do not change the base files.

compatibility mode

If you configure your project to use a version of ePublisher that is earlier than the version that is installed on your computer, you run 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 ePublisher installation files. When you customize a file, a copy of the installation file is created in the **Targets** and **Formats** folder in your project. In this article, customizations are called *overrides*.

installation files

The ePublisher files that are located in the `\\Program Files\WebWorks` folder. They are also called *base files*.

CAUTION: Do not change the installation files.

overrides

Changes that you make to the ePublisher installation files. When you create an override, a copy of the installation file is added to the **Targets** or **Formats** folder in your project. Overrides are also called *customizations*.

super customization

A minimal override that isolates custom XSL code, and that enables you to override an XSL file without mixing custom code with installation code. Super customization is also called *wwtransform:super*.

wwtransform:super

An extension to ePublisher URI Resolvers that extends the tiered lookup behavior to enable you to import the lower-priority version of the file in addition to specializing some aspect of it in a higher version. The wwtransform:super is 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 gives me an idea of what overrides I can expect to keep and what I can expect to remove 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.

Formats versus Targets

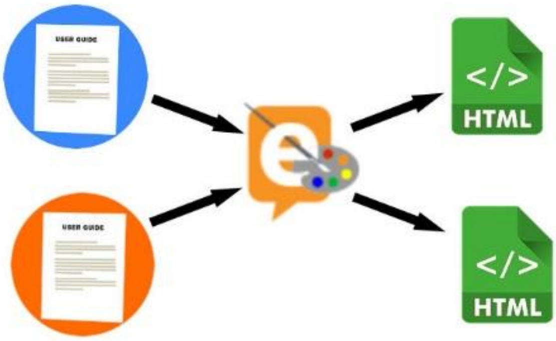
To create and maintain your WebWorks projects, it is important to understand formats and targets, how to use them, and the differences between them. In addition, the concept of formats and targets becomes more complex if you use the *Considerations for the Reverb skin formats*.

Format is the output type that is generated by ePublisher. For example, WebWorks Reverb 2.0 or PDF.

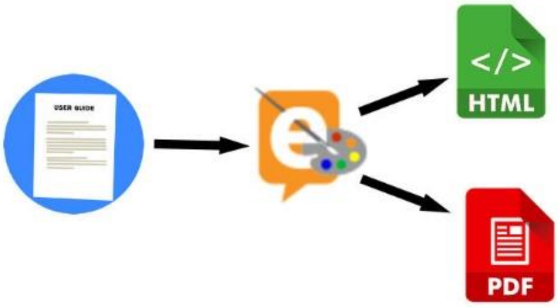
Target is your goal-specific output. You can have one or more targets in a project. Each target might be for a different product, but use the same output format, or each target might be for the same product, but use a different output format

TIP: Your targets appear in the Active Target drop-down list on the ePublisher toolbar.

Use case: You have two products, Product A and Product B, that use one type of output format. The input files for each product are different and the formatting for each product varies a little, but the output format is the same. In this case, you could have a target called *ProductA*, and another target called *ProductB*.



Use case: You have one product that requires two different output formats, such as WebWorks Reverb 2.0 and PDF. In this case, you could have a target called *Reverb 2.0* and a target called *PDF*.



Formats folder versus Targets folder

CAUTION: DO NOT customize the installation files in \\Program Files\WebWorks. Copy the files to your project.

When you create overrides to the ePublisher installation files, you usually create the override file in the **Formats** and the **Targets** folders. The Formats and Targets folders are generated automatically when you create overrides from the *Description of the Advanced menu* tabs.

- Overrides in the Formats folder apply to the type of output, such as the WebWorks Reverb 2.0 or the PDF-XSL-FO output.
- Overrides in the Targets folder apply to the specified active target output, such as ProductA versus ProductB.
- You can have both a Formats folder and a Targets folder in your project.
- If overrides conflict, then overrides in the Targets folder take precedence over overrides in the Formats folder.

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

In some cases, you must create your overrides manually. For more information, go to the topic *Creating Project Target Overrides* on www.webworks.com.

Considerations for the Reverb skin formats

The WebWorks Reverb and WebWorks Reverb 2.0 formats are treated as formats or targets, depending on how you use them. In the installation folder, the **Formats** folder contains plugin folders for each of the Reverb formats. These plugins are also called skins.

Initially, when you create a project that uses a Reverb format, ePublisher uses a default skin. In this case, it is considered a format. ePublisher uses the following skins as a basis for the default Reverb skins:

- WebWorks Reverb 2.0 uses the Neo skin
- WebWorks Reverb uses the Classic skin

However, if you select a skin from the **Targets Settings** dialog box, then the output is considered to be a target.

Technically, the skins are built-in overrides to the default format. Each plugin folder has files that are overrides to the files in the installation folder. It is important to understand, however, that different plugin folders do not necessarily contain the same override files. For example, in the WebWorks Reverb 2.0 format, the Corporate skin has an override for the `Connect.asp` file, but the Classic skin does not.

When you create an override for a skin on the **Manage Format Customizations** tab, ePublisher copies the file from the installation folder.

When you create an override for a skin using the **Manage Target Customizations** tab (assuming that you have not created an override in the equivalent format file), the file is copied from the plugins folder *providing that there is a matching file for that skin*. If there is no matching file in the plugins folder, then the file is copied from the installation folder. So in our previous example, the `Connect.asp` file would be copied from the plugins folder for the Corporate skin, but from the installation folder for the Classic skin.

IMPORTANT: If you change the skin in your target, review your customizations carefully.

Your project might be more complicated and include format AND target overrides for a skin. The following table describes where the files are copied from depending on the type of override you create.

Manage Customizations tab	Override consideration	Copied file
Target	No matching file in the skin plugin folder	Installation folder
Target	Matching file in the skin plugin folder	Skin plugins folder
Target	No matching file in the skin plugin folder, but an override already exists on the Format tab	The file that was already used for the format override (that is, the file in your project Format folder) is used for the target override
Format	No target overrides for the skin file	Installation folder

If you have multiple targets that use the same skin, it is good practice to make as many overrides as possible at the format-level. Any overrides that need to be made to the same files at the target-level should be made after all the format-level changes are made. This should help minimize the number of overrides that you need to maintain. However, if you must make overrides at the target-level that do not have overrides in a matching format-level file, you will have to make the same overrides for every target that you have.

> **TIP:** For WebWorks Reverb 2.0, the default skin is the Neo skin. If you want to create multiple targets using the Neo skin, you can minimize maintenance by using the default skin.

Description of the Advanced menu

The **Advanced** menu helps you to manage your overrides by using the ePublisher interface. You can access override options from either the **Advanced** menu on the toolbar or by right-clicking a file **Manage Target Customizations** or the **Manage Format Customizations** tab.

The following table describes the options that are available on the **Advanced** menu.

Item	Description
Manage Format Customizations	Opens a tab that displays the files that you can override 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 override • edit override files • compare the override files to the installation files • remove the override files from your project For information about the appearance of files on the **Manage Format Customizations** tab, go to *Types of overrides*.
Manage Target Customizations	Opens a tab that displays the files that you can override 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 override • edit override files • compare the override files to the installation files • remove override files from your project For information about the appearance of files on the **Manage Target Customizations** tab, go to *Types of overrides*.
Explore To Customizations	Opens Windows File Explorer to the folder where the override file is located.

Item	Description
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. This option is not available on the **Advanced** menu unless the file has already been added as an override. However, you can open and edit any file on either of the **Manage Customization** tabs 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. You can right-click the file, and then click **Remove Customization**.
View File Differences	Opens the selected file and the equivalent installation file in the file comparison tool 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 generated 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.
Create Super Customization	Creates a skeletal transform that is configured to enable you to 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 your override file from your project. When you regenerate your output, ePublisher uses the file that is in the installation folder.

Item	Description
Configure Tools	Opens the **WebWorks ePublisher Preferences** dialog box. The **Editor Preferences** tab enables 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 in Windows. The **Diff Preferences** tab enables you specify which file comparison tool to use when you compare your override files to the installation files. In addition, you can configure these options by going to **Edit > Preferences**.

Types of overrides

Depending on where you place your overrides, they can affect your project in the following ways.

Override type	Override placement	Visibility	Example
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
Project wide	\\Formats\[override]	Not visible	\\Formats\Adapters

The following table describes what the appearance of the files on the **Manage Customizations** tabs means.

Appearance	Meaning
Bold	The override has an equivalent installation file and the file path is mirrored in the **Targets** or **Formats** override folder.

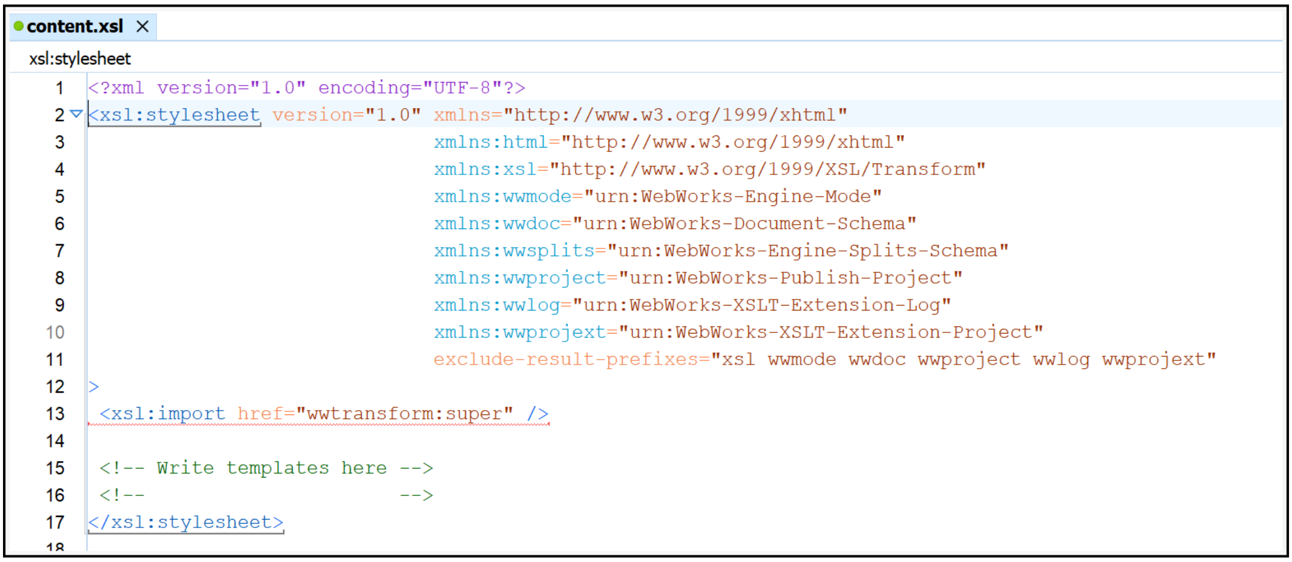
Appearance	Meaning
Blue	Folder and file names appear blue in the **Manage Target Customizations** or **Manage Format Customizations** tab under the following conditions: • The override has an equivalent installation file, but the file path is not mirrored in the target or format override folder. For example, the files in the **Shared** folder. • The override file does not have an equivalent installation file, but is located in the **Targets** or **Formats** folder. You cannot add these override files to your project using the Advanced menu; you must add them manually using Windows File Explorer. In addition, you must compare file differences from Windows File Explorer. **Example:** If you want to create an override to the `locales.xml` file, then you must copy the **Shared** folder, subfolders, and the `locales.xml` file into your **Targets** or **Formats** folder manually. The file path to the override file does not mirror the file path to the equivalent installation file. • Installation file path: `C:\Program Files\WebWorks\ePublisher\2019.1\Formats\Shared\common\locale\locales.xml` • Override file path: `C:\[project_name]\Formats\WebWorks Reverb 2.0\Shared\common\locale\locales.xml`
Gray	**Applies to:** WebWorks Reverb 2.0 and WebWorks Reverb format-level overrides To understand this explanation better, review the topic *Considerations for the Reverb skin formats.* Under certain circumstances, the files on the **Manage Format Customizations** tab might appear as follows : • If the selected skin has a target override in the file that matches the format file, but the format file does not contain any overrides, the format file appears gray. • If the format file contains an override, and the matching target file in the skin also contains an override, the format file appears gray and bold. In both of these cases, the format files are ignored when the output is generated, whether or not it contains an override. The target override file is used instead.

Appearance	Meaning
Not visible	The override file is located at the top level of the **Formats** folder, but not within the folder of the specified format. For example, `\\Formats\Adapters\xml\scripts\dita\wwdefault.config`. **NOTE:** For an example of when you might use the **Adapter** folder in a project, see the topic Mapping DITA Classes to ePublisher Styles in the WebWorks online help. You cannot add or manage these types of overrides using the **Advanced** menu; you must add and manage them manually.

Super customizations

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

When you use the **Advanced** menu to create a super customization, a skeletal transform is created that is configured so that you can create overrides using the wwtransform:super mechanism.



```
<?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 only invoked if you use <xsl:apply-imports>. This enables you to process the code by using the templates in the underlying stylesheet.

Super customizations can only be applied to XSL files.

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

The following WebWorks wiki articles describe use cases that use super customizations.

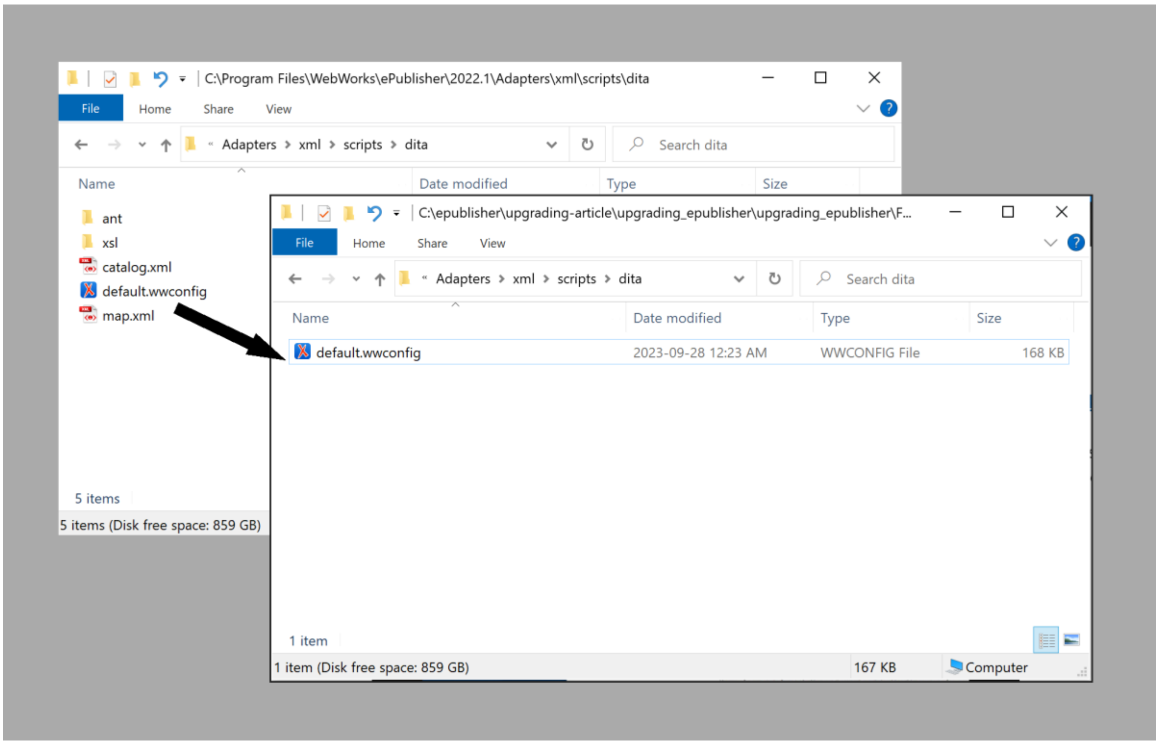
- *2-Celled Note Table*
- *Custom Marker Processing*

Best practices for creating overrides

I strongly recommend that you develop some basic housekeeping habits when you create your overrides. Doing so will help you immensely when you upgrade ePublisher in the future. In addition, if someone else takes over your project, they will be very grateful to you.

Take only the files that you need

When you create overrides *manually* for your project, copy only the files that you need to customize. For example, if you need to customize the `default.config` file, copy only that file; do not copy all of the files in the same folder.



C:\Program Files\WebWorks\ePublisher\2022.1\Adapters\xml\scripts\dita
File
Home
Share
View
Adapters › xml › scripts › dita
Search dita
Name
Date modified
Type
Size
ant
xsl
catalog.xml
default.wwconfig
map.xml
5 items
5 items (Disk free space: 859 GB)
C:\epublisher\upgrading-article\upgrading_epublisher\upgrading_epublisher\F...
File
Home
Share
View
Adapters › xml › scripts › dita
Search dita
1 item
1 item (Disk free space: 859 GB)
167 KB
Computer

Name	Date modified	Type	Size
default.wwconfig	2023-09-28 12:23 AM	WWCONFIG File	168 KB

Regardless of whether you create a **Targets** or **Formats** folder in your project, mirror the folder hierarchy that is in the installation folder. For example, for a project-wide override, the file path should be structured in the following way:

Installation folder: `C:\Program Files\WebWorks\ePublisher\2022.1\Adapters\xml\scripts\dita`

Project folder: `[your project]\Formats\Adapters\xml\scripts\dita`

Add comments to your code changes

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

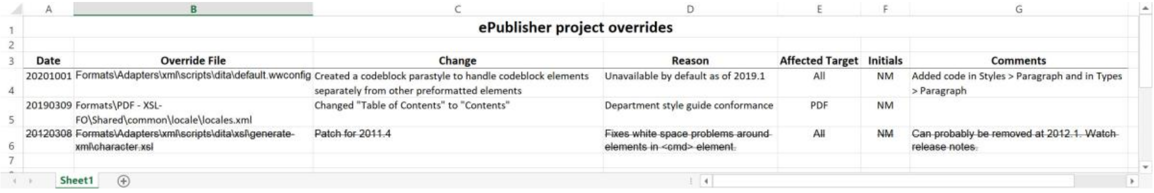
```
VarPDFURL = VarDocumentParentURL + "/" + VarPDFLink;
// 20190503 NM changed the following line to force PDFs to open in a
separate window //
window.open(VarPDFURL, "WWHelp_PDf_Window");
```

Create detailed records of your overrides in an external file

Keep a record of your overrides in an external file, such as a txt file or spreadsheet, that lists which files you changed and include a brief description of the changes that 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.

I like to 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. In addition, when I no longer need an override, I still keep the information about it with a note about when I removed the override and why.

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



ePublisher project overrides

Date	Override File	Change	Reason	Affected Target	Initials	Comments
20201001	Formats\Adapters\xml\scripts\dita\default.wwconfig	Created a codeblock parastyle to handle codeblock elements separately from other preformatted elements	Unavailable by default as of 2019.1	All	NM	Added code in Styles > Paragraph and in Types > Paragraph
20190309	Formats\PDF - XSL-FO\Shared\common\locale\locales.xml	Changed "Table of Contents" to "Contents"	Department style guide conformance	PDF	NM	
20120308	~~Formats\Adapters\xml\scripts\dita\xsl\generate-xml\character.xsl~~	~~Patch for 2011.4~~	~~Fixes white space problems around elements in <cmd> element.~~	All	NM	~~Can probably be removed at 2012.1. Watch release notes.~~

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.

Comparing your overrides to the installation files

TIP: 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 tool 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).

Make sure to watch the WebWorks video about comparing files and reducing your code differences.

Using the Advanced menu to compare folders and files

Before you use the **Advanced** menu to compare your override files to the installation files, you must configure ePublisher to open your file comparison tool. You can do this on the **Diff Preferences** tab of the **WebWorks ePublisher Preferences** dialog box.

In this task, you use the **Advanced** menu and the **Manage Customizations** tabs to open your comparison tool and compare your override folders and files to the installation folders and files.

> **NOTE:** If you must compare your folders and files manually, you can do so using a combination of Windows File Explorer and your comparison tool.

1. Configure ePublisher to locate the correct installation folder to look in when you compare your overrides.
    a. Go to **Projects > Project Settings**.

       The **Project Settings** dialog box appears.
    b. On the **General** tab, in the **Base format version** field, 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 tool.

> **TIP:** If you decide not to perform this step, your file comparison tool 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 tool.

2. Go to **Advanced**, and then choose the appropriate option.
   - To display which files contain overrides for the selected target, click the **Manage Target Customizations**.
   - To display which files contain overrides for the selected format, click the **Manage Format Customizations**.

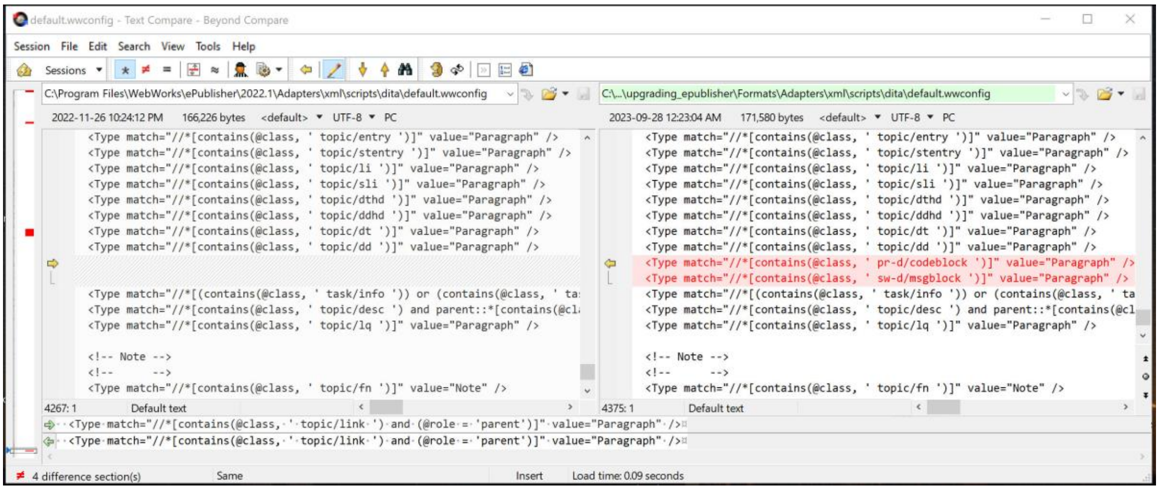
   A tab appears that displays the files that are available for customization. The files that contain overrides appear in bold font.

Many comparison tools enable you to 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 that you do not have to rerun the comparison.

Reducing 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 override files. If you are not familiar with what the new code is doing, this task can be daunting. 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.



default.wwconfig - Text Compare - Beyond Compare
Session File Edit Search View Tools Help
Sessions

```
C:\Program Files\WebWorks\ePublisher\2022.1\Adapters\xml\scripts\dita\default.wwconfig
2022-11-26 10:24:12 PM   166,226 bytes   <default>   UTF-8   PC
    <Type match="//*[contains(@class, ' topic/entry ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/stentry ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/li ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/sli ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/dthd ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/ddhd ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/dt ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/dd ')]" value="Paragraph" />


    <Type match="//*[(contains(@class, ' task/info ')) or (contains(@class, ' ta
    <Type match="//*[contains(@class, ' topic/desc ') and parent::*[contains(@cl
    <Type match="//*[contains(@class, ' topic/lq ')]" value="Paragraph" />

    <!-- Note -->
    <!--      -->
    <Type match="//*[contains(@class, ' topic/fn ')]" value="Note" />
4267: 1   Default text
```

```
C:\...\upgrading_epublisher\Formats\Adapters\xml\scripts\dita\default.wwconfig
2023-09-28 12:23:04 AM   171,580 bytes   <default>   UTF-8   PC
    <Type match="//*[contains(@class, ' topic/entry ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/stentry ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/li ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/sli ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/dthd ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/ddhd ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/dt ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' topic/dd ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' pr-d/codeblock ')]" value="Paragraph" />
    <Type match="//*[contains(@class, ' sw-d/msgblock ')]" value="Paragraph" />
    <Type match="//*[(contains(@class, ' task/info ')) or (contains(@class, ' ta
    <Type match="//*[contains(@class, ' topic/desc ') and parent::*[contains(@cl
    <Type match="//*[contains(@class, ' topic/lq ')]" value="Paragraph" />

    <!-- Note -->
    <!--      -->
    <Type match="//*[contains(@class, ' topic/fn ')]" value="Note" />
4375: 1   Default text
```

```
  <Type match="//*[contains(@class, ' topic/link ') and (@role = 'parent')]" value="Paragraph" />
  <Type match="//*[contains(@class, ' topic/link ') and (@role = 'parent')]" value="Paragraph" />
```

> **IMPORTANT:** Perform this task 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.

1. Back up your override files to a different location on your computer.
2. Compare your override files to the installation files.
3. Decide whether 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 the new code in the installation file.
    - If WebWorks has not changed the code and you still need your override, keep your override code.
4. If WebWorks has changed the code or added new code to the installation file in general, and you cannot tell what the code does, then go to What the heck is *that* code doing? or *Three-way* merge.
5. After you reduce all the differences between your override files and the 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, go back and compare your customized files to the installation files again.
6. When you are satisfied that all the differences have been reduced, update your stationery.

What the heck is that code doing?

> **REMEMBER:** Before you reduce the differences, make sure that you back up your override files to another location on your computer.

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.

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.

1. Reduce the code differences 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, then copy your backup override 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, make the changes one file at a time. Or if there is a defined section within the file, then make the changes in that section.
8. Regenerate your project in ePublisher.
9. Continue to regenerate your project and add new code until all of the new code is added to your override files and your output generates as expected.

Three-way merge

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

One benefit of this is that it enables 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\2019.1\Formats\WebWorks Reverb
2.0\Pages\Page.asp
```

```
C:\Program Files\WebWorks\ePublisher\2022.1\Formats\WebWorks Reverb
2.0\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 tool to perform a three-way merge, if your comparison tool 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, go to the WebWorks blog *Wrangling the 3-way merge*.

Acknowledgments

Thank you to:

- Ashton French of WebWorks for patiently helping me to bring this article up to date.
- Ben Allums, formerly of WebWorks, for providing the information for this article, reviews, arguments, and encouragement.
- Jesse Wiles, formerly 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 the **Advanced** menu. • Noted differences between managing overrides manually and using the **Advanced** menu. • Added content about super customizations.
C	August 2019	2019.1	• Updated and revised the content. • Added content about Reverb skins. • Removed information about pre-2012.2 version of ePublisher.
D	October 2023	2022.1	• Made editorial changes. • Updated images and file path dates. • Made cosmetic changes.