Merged WebHelp and HTML5 Help

What's covered?

This topic is about how to create merged WebHelp using **Classic** versions of RoboHelp up to and including RoboHelp 2019. The process is the same for HTML5 layouts but see Item 2 in Snippets before you proceed.

See Merged Help New UI if you are using 2019 New UI or above.

The process for CHMs is different has remained constant going back many versions. See Merged Microsoft HTML Help (CHMs) as well as this page.

The feedback that I have had tells me that this is regarded as the simplest method around. I use it for a merge that at one time involved some 14,000 topics in over 700 folders, so it's also robust! I have also been told that Adobe Technical Support point users to this page!

At one time it was not possible to set up a merge within a merge but that seems to have changed somewhere along the way. I suspect it was as the search algorithm changed.

In Download the demo below you can download a sample project.

The Steps

How this method works

According to the online help for RoboHelp, the basics of merged WebHelp are that in TOC of the parent you include placeholders for the child projects. When you generate the WebHelp for the parent, a folder named mergedProjects will be created and that will contain folders to which you generate the child projects. All quite simple and correct, as far as it goes.

What is not explained is that unless your source projects are set up with the correct structure, creating cross project links will require you to work out the relative path in the output and type that into the HTML. Fortunately it is not difficult to set things up so that RoboHelp creates the links for you so that you can rely on them working in your outputs..

The two key elements to the method described here are:

- The source projects are maintained within a simple structure.

- The parent default topic contains a redirect that opens a designated topic in a child project.

When merged WebHelp is opened from the start page, what you normally see is shown in Figure 1. The start page opens the tri-pane window and the parent project's default topic is displayed.

Figure 1. The conventional way of viewing merged WebHelp

What this method does is open the help via the parent project, as it must, but the redirect means the topic displayed is in fact from one of the child projects. What the user sees is shown in Figure 2:

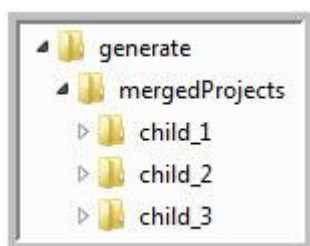**Figure 2. The user sees a topic from one of the child projects, rather than the parent.**

In reality, the content of the topic from the child project would be the same content that would have been in the parent so the user sees no difference.

As all the content is in child projects, creating cross-project links can be done via the RoboHelp interface and that avoids broken cross project links. That and what you need to know about merged help is described in the following sections.

The Structure

The folder structure of the generated and published files is the same and is shown in Figure 3. The output from the parent is at one level in the "generate" folder, then there is a "mergedProjects" folder that is a "container" for folders that hold the output from the child projects.

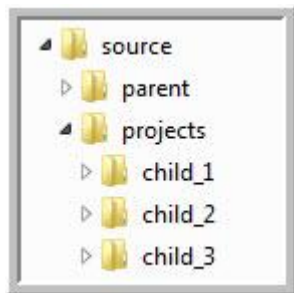**Figure 3. Generated and published files folder structure**

The same structure is needed for your source projects so that you can create links between them. With this structure, links can be created using the RoboHelp interface. The only wrinkle is that links between parent and child will still break in the output. To handle that using this method, the content that is normally placed in the parent is placed in child_1 and the parent is effectively just a shell, albeit very important.

The parent project in this setup contains only one topic that the user will never see. When the help is opened by calling the start page of the parent project, what happens is that the toolbar and navigation pane display, then the default topic for the parent project starts to open but a redirect in it calls the default topic for child_1. That is what the user sees.

So that you can set things up easily when working on the source files, you need the folder structure shown in Figure 4 for your projects.

Figure 4. Use this model structure for your merged help source projects.

While you can have content in your parent project, it is much simpler if all content is contained in the child projects so that the parent is effectively nothing more than a shell. If you do have content in the parent, avoid links between the parent and any child, they will break when you generate the output.

Download the Demo

A ZIP file containing a working example can be downloaded. I have set all the file dates and times to be the same. This will enable you to identify any files you change. The ZIP file must be extracted to retain the folder structure.

The demo was created in RoboHelp 9 so users of later versions should allow RoboHelp to upgrade the projects. No other changes are required.

When you open the projects, you will see a warning about external links, just click OK.

I strongly recommend that you download the demo and work on it first before attempting to work with your live project(s). The demo is intentionally kept simple so that you can easily see what is going on.

As always, before you start using a new method on your live projects, back them up first.

Merge within merge demos

RoboHelp 9 Merge within merge

RoboHelp 10 and above Merge within merge

In the RoboHelp 9 demo, when viewing a topic in the lower level merge (referred to as grandchildren), the parent of that merge will show in the breadcrumbs. It will not show in the RoboHelp 10 and above breadcrumbs. You will see that I have renamed the breadcrumb of the lower level merge parent in RoboHelp 9 so that the user does not see "Home" twice.

Search for Redrabbit to see that the search does find content in the lower level merge.

The table of contents

You need to consider how you want the TOC to be structured.

- You cannot have the book for one child project nested within the book for another child project. That is a limitation of merged WebHelp generally rather than of this method.

- You can nest a child project within a book in the parent project but because this method works by putting all the content in projects at child level, there is no book in the parent TOC into which you can nest a child project. The TOCs for each child will appear one under the other.

That works with no problems in the merges I have seen. The alternative is hand coding all your cross project links. Not recommended.

There is another method of merging that does give you the ability to nest a child project within a book in the parent project. However, as described in another article, it is a more complex method and I do not recommend it unless you are hell-bent on this requirement. It is a more time-consuming method and prone to errors.

Step 1: Create the structure

Create a structure for the source projects similar to the one shown in Figure 4 which is repeated here.

You can call the folders whatever you want. I simply use "parent" and "projects" so that alphabetically the order is the parent first and then the children.

Step 2: Create the parent project

1. Create a WebHelp project in the parent folder you created in Step 1. I have used "parent" as the folder name and the project name in the demo. Avoid spaces in your project names.

2. Open the default topic that RoboHelp creates and remove all text, including the topic heading.

3. Select the skin you require and edit that as necessary. Remember, it is this skin that you will see for the whole merged output. Any skin for a child project is only seen if that project is opened independently of the merged output.

4. If any of your child projects will have a browse sequence, it is essential that you create one for this project. I know there is only one topic that can go in it. The user will never see this browse sequence, but unless you create it, the checkbox to enable browse sequences will be disabled when you generate the help. If the parent browse sequence is not enabled, the child browse sequences will not work.

5. For now, close this project.

Step 3: Create the child projects

First, create the main child project in the child_1 folder you created in Step 1.

This project will contain what would otherwise have been in the parent project. Typically, this will be an introduction to the product, how to use the help, and other such topics. The default topic for this project is the one you will later set up to be the "default" for the merged help.

Then, create any other child projects you require in the folders for child_2, child_3 and so on.

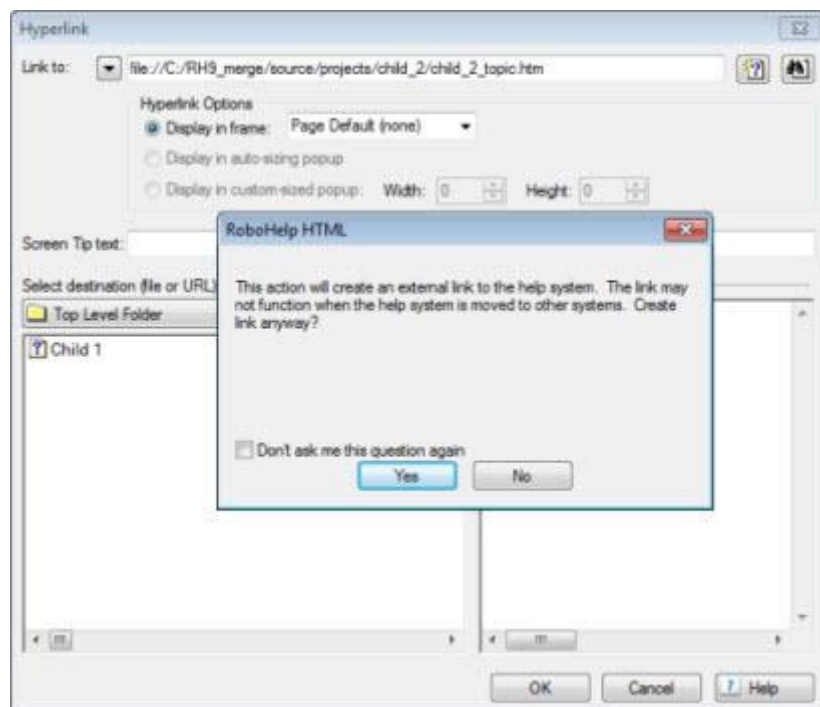## Step 4: Develop your help and create links

To create links between the child projects:

1. Highlight the text that will be the link.

2. Select Insert > Hyperlink or the hyperlink icon.

3. Click the "Link To" dropdown shown in Figure 4 and select File.

4. Browse to and highlight the required file and click Open (or double click the required file). The absolute path will be shown in the "Link To" field. There is no need to amend it. After you click

OK, RoboHelp will amend the path to a relative path (double click the link if you want to check this).

When creating links between projects, you may get a warning about links to external files. That is normal, and I have selected the "Do not ask me the question again" option. You can restore the warnings in File or Tools > Options (depending on RoboHelp version).

Figure 5. "Link to" is where you create links. You may see the warning shown.

Note re Unix
When you publish a single WebHelp project with the "Use Lowercase Filenames" option selected, RoboHelp changes both the filenames and any links to lowercase.
When you publish each project in a merged help setup with the "Use Lowercase Filenames" option selected, RoboHelp will change all the filenames to lowercase, but links to files will only be changed to lowercase if the link is to a file in the project you are publishing. So, all the filenames will be lowercase, but some of the links will still be mixed case. This mismatch causes broken links on a Unix box.
You must ensure the case is the same for the files and links in your output. I ensure that all file names are lowercase from the start.
This mismatch is not an issue on Windows systems.
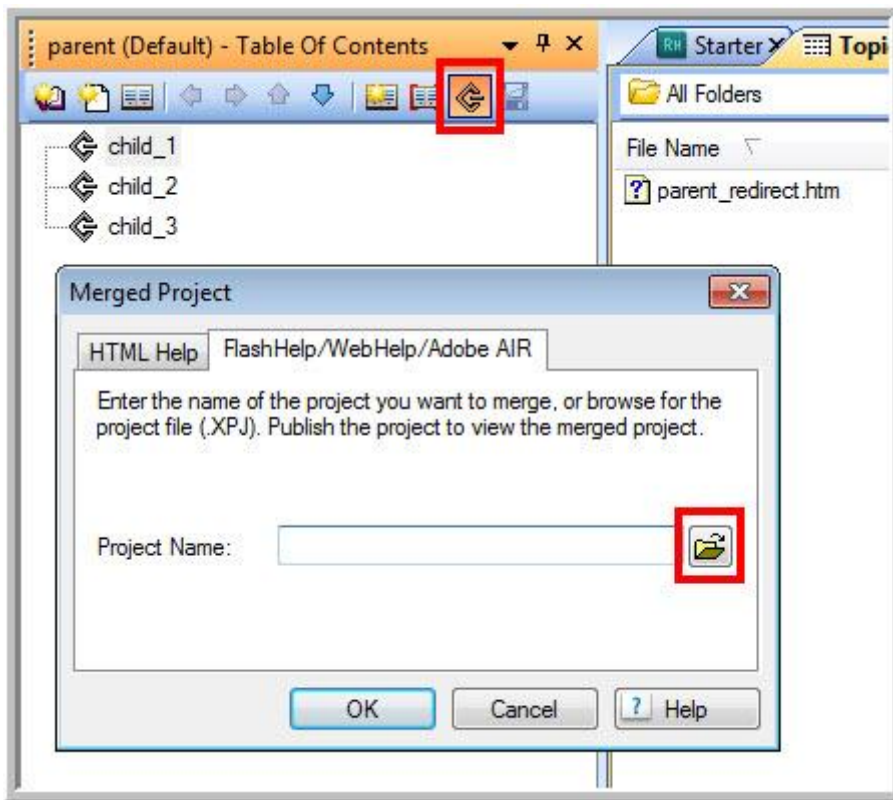

Step 5: Update the parent project

The next step is to create the references to the child projects and add the redirect in the only topic in this project. Reopen the parent project.

Set up the parent project TOC

The parent project has no TOC as such, but this is where you tell the parent what children it has! Here we follow the RoboHelp instructions for including the TOC from the child project(s).

1. Click the Insert Merged Project icon (highlighted in Figure 6)

2. The Merged Project dialog box will appear. Make sure the FlashHelp/WebHelp tab is displayed. The project name will be blank at this point.
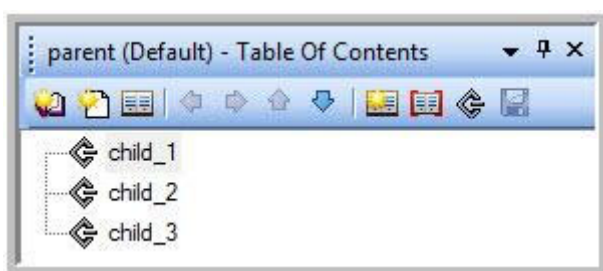
Figure 6.

3. Click the folder icon and navigate to the XPJ file for the child project and click Open.

4. Select a child project from the Open dialog box.

Figure 7

5. Repeat steps 1–4 until all the child projects are shown in the TOC as shown in Figure 8. The order of the references is the order in which their TOCs will appear in the merged help.

Figure 8. The child projects shown in the parent project's TOC.

Add the redirect
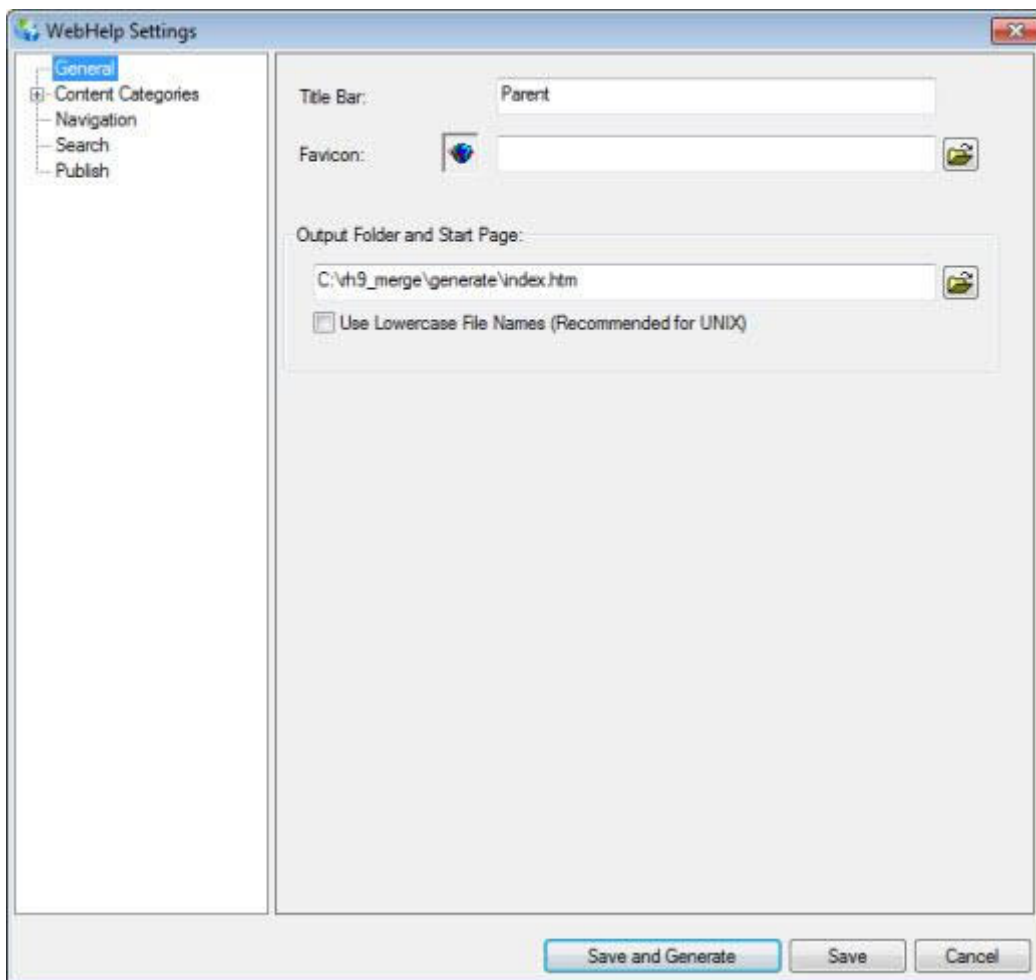
The simple method for a redirect is something like:

```
<meta http-equiv="refresh" content="0;URL=./mergedProjects/child_1/child_1_topic.htm" />
```

However, this will not work in Firefox at least. Instead use this method.

1. Click in the box below, press CTRL + C and paste the following above the </head> tag. You will need to amend the path to point to the topic in Child 1 that you want to be the default.

```
<script language="JavaScript" type="text/javascript">//<![CDATA[
<!--Script courtesy of http://www.web-source.net - Your Guide to
Professional Web Site Design and Development //-->
var time = null
function move() {
window.location = './mergedProjects/child_1/child_1_topic.htm'
}
//]]></script>
```

Note that it is a condition of using the script that the acknowledgement remains in it.

3. Amend the body tag as below. Change the timeout if required, it is set in milliseconds. If your body tag already has an onload event, add a semi colon after it and do not repeat the onload= part.
   ```
   <body onload="timer=setTimeout('move()',100)">
   ```

Background color

It is important that the topic to which the user will be redirected and the parent topic have the same colored background. If the two topics have different colors, the redirect may be detected.

Step 6: Generate and publish the parent project

You can generate to the !SSL! folder within your parent project, but I recommend that you generate to a folder outside the project; it avoids a lot of confusion and some problems that regularly crop up on the forums. In the download project, I have set things up to generate to a folder called generate. You can name it whatever you prefer.

You do not have to publish each time you generate and you can skip step 6 in this section for now if you wish. I will explain more about publishing later.

1. Start the Single Source Layout for WebHelp so that you see the configuration settings. By default it will appear as in Figure 9.

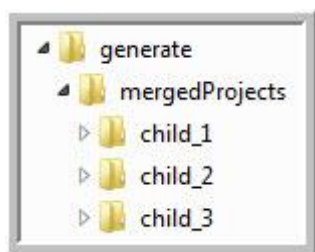   **Figure 9. The Single Source Layout for WebHelp.**

2. Change the Output folder and filename. The folder I have selected is outside the project folders. It's cleaner that way, with source in one folder and the output quite separate. The folder must be on your local drive.
   Starting from RoboHelp 9, the default start page file name will be index.htm in new projects, if you see anything else I suggest you rename to index.htm. That is a standard name for a website and one familiar to your developers. It helps them identify the start page. Notice that I have not selected Use Lowercase File Names (see Figure 8). As already explained, in a merge that may result in broken links. If you want lowercase filenames, create your topics with lowercase file names rather than using this option.

3. If your default browser is Internet Explorer, I suggest you tick the Add Mark of the Web option while you are setting things up. You will find that in the Navigation Settings.

4. Leave everything in the Content section unchanged.

5. Take a look at the other settings but I suggest you leave the defaults unchanged for now. Once you are familiar with merged WebHelp, you can use whatever options you require.

6. Publishing is optional and not everyone needs to publish. If you are not sure, see Step 11: Delivery. If you do need to publish, click New and enter the required location in the New Destination dialog box. The name can be whatever you want and it can be a local drive.

7. Click Save and Generate and the Parent project will be generated.

8. When RoboHelp has generated the project, if you have set up a Publish location then the Publish button will be enabled.

Step 7: Generate and publish the child projects

When you generated the parent project, it will have created a structure like the one in Figure 3, which is repeated here. The child project folders will be empty at this stage other than a few placeholder files that RoboHelp creates. Ignore them.

Follow the same steps as for generating and publishing the parent, except that you point to the appropriate child folder; for example,

C:\rh9_merge\generate\mergedProjects\child_1. Don't forget you also need to check the browse sequence options for the relevant child projects.

Care re merged**P**rojects

In the above example, you will see that the "P" in mergedProjects is uppercase. I recommend that you do not change this. Sometimes developers will insist on all lowercase paths and filenames, and if you encounter this you should explain that the capital "P" is forced by RoboHelp and that the workaround of a mass Find and Replace risks breaking the help.

Step 8: Forcing the TOC to synchronize

Provided you set the Synchronize TOC option to Automatically, once the help has been opened, the topics displayed should synchorize with the TOC. You may need to set Optimise Speed to Local PC or Intranet.

Step 9: Check for broken links

RoboHelp does not check external links, which includes links between the child projects, so you may want to consider a website link-checking utility such as Xenu.

Step 10: Test

Go to either your rh9_publish folder (or your rh9_generate if you skipped publishing) and double-click the start page; index.htm if you followed my example. Watch very carefully: what happens is that the default topic does open but immediately redirects to the default topic for child_1. You will probably not even notice it.

If you want to test that, apply a bright red background to the parent default topic, then you will see it! That is also why I suggested earlier giving this blank topic the same color background as the "default" topic to help mask its transitory appearance.

The parent default topic cannot be accessed, as there is no TOC for the parent project, you did not index that topic, and there is no text the search engine can find.

There it is: merged WebHelp with easily created links and all ready for delivery.

Step 11: Delivery

Generating and publishing: what's the difference?

Generating is about RoboHelp converting the content of your source files to HTML code that can be interpreted by different browsers.

Publishing is about taking those output folders and files and putting them onto a server.

Click here for more detailed information about Publishing

What if all the projects are not required in a particular installation?

Take a copy of the full merged WebHelp output and delete from the copy the folders for the projects that are not to be installed. It really is that simple.

Take a copy of rh_generate from the download, delete the folder for Child 3, and then open the help. You will see that the TOC no longer shows the book for that project and the index no longer shows its topics. Also, any links from topics to the deleted child will break. You need to consider that in your design. In some cases, it may mean you do need to generate a new output after removing those links from the source.
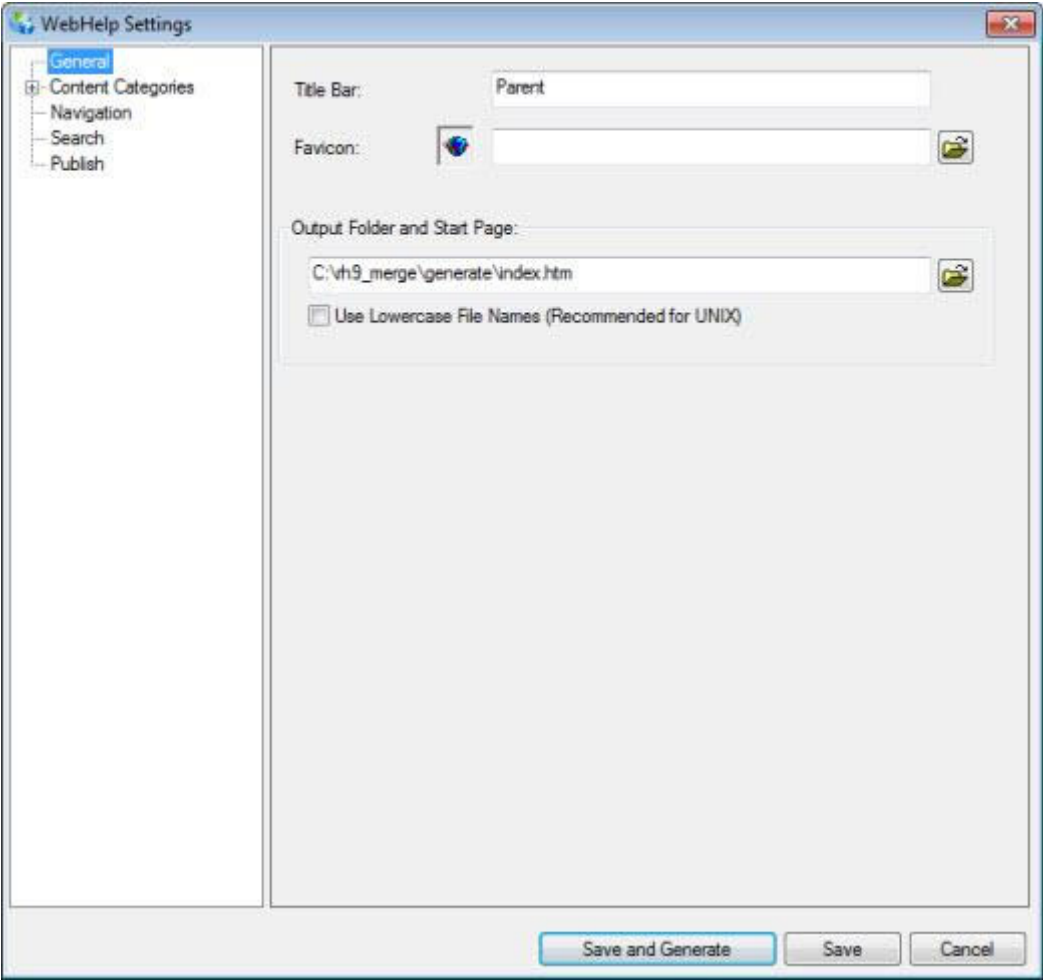
RoboHelp Output Configuration

When you open the Output SSL, you will see the dialogs shown below.

Remember, you have to go through the settings of each project in the merge.
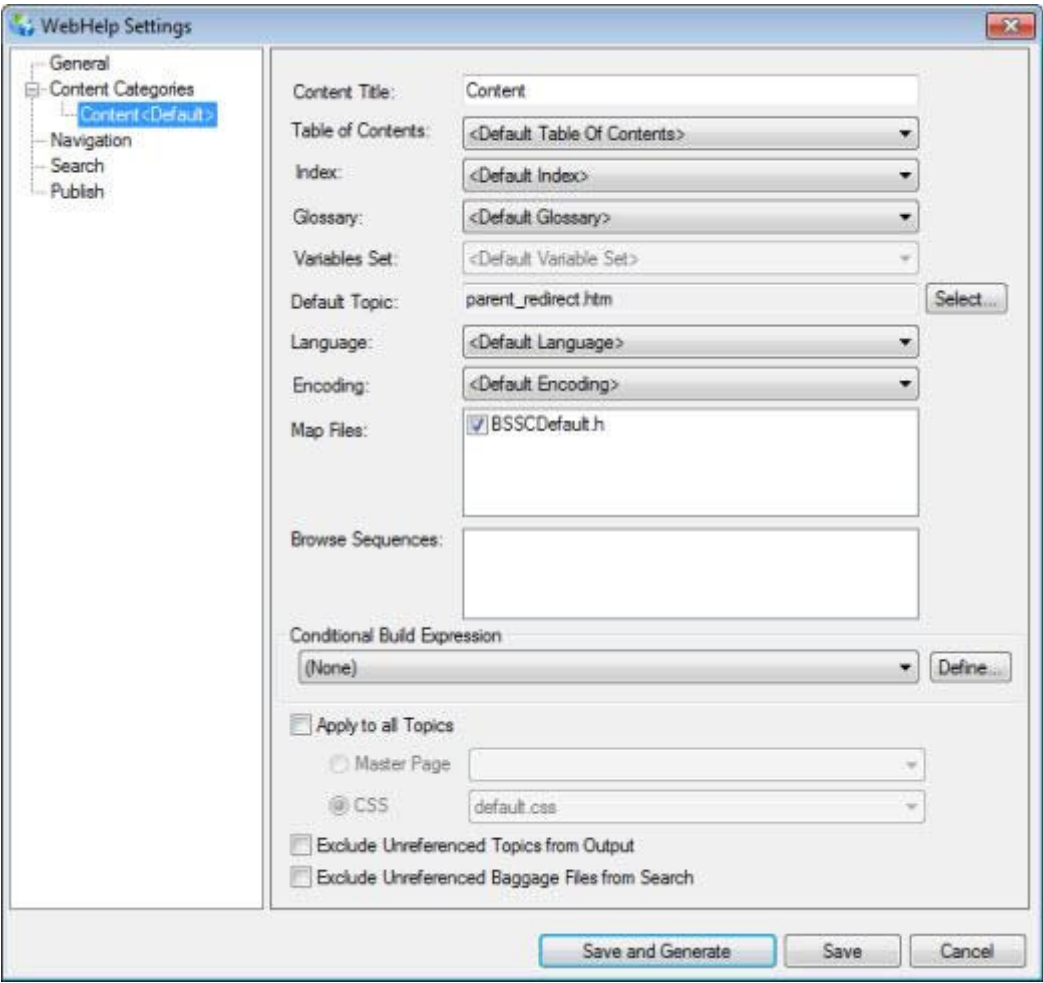
General Settings

Figure 10.

Field	Information
Title Bar	What appears in the browser title bar used to be based on what was defined in Project Settings. Now you can set it here.
Favicon	If you want an icon to appear in the browser tab, set the icon here. It must be a 16 X 16 png or ico file. Note the path is absolute and based on a drive letter so the link will likely break if you move the project to a different machine with different mappings.
Output Folder and Start Page	Don't confuse the Start Page with your Default Topic. They are not the same thing. The Start Page is what opens the WebHelp. The Default Topic is what displays in the topic frame.

User Lowercase File Names	The Use Lowercase File Names option is not recommended for merged help. It will force all filenames in the project being generated to lowercase, and it will change the links to topics within that project to lowercase. It will not change the links to topics in other child projects. When you generate those, this option will change the filenames there, so there will be a mismatch between link and topic. If the help is on a Unix box, that will cause a broken link.

Content Settings

Figure 11.

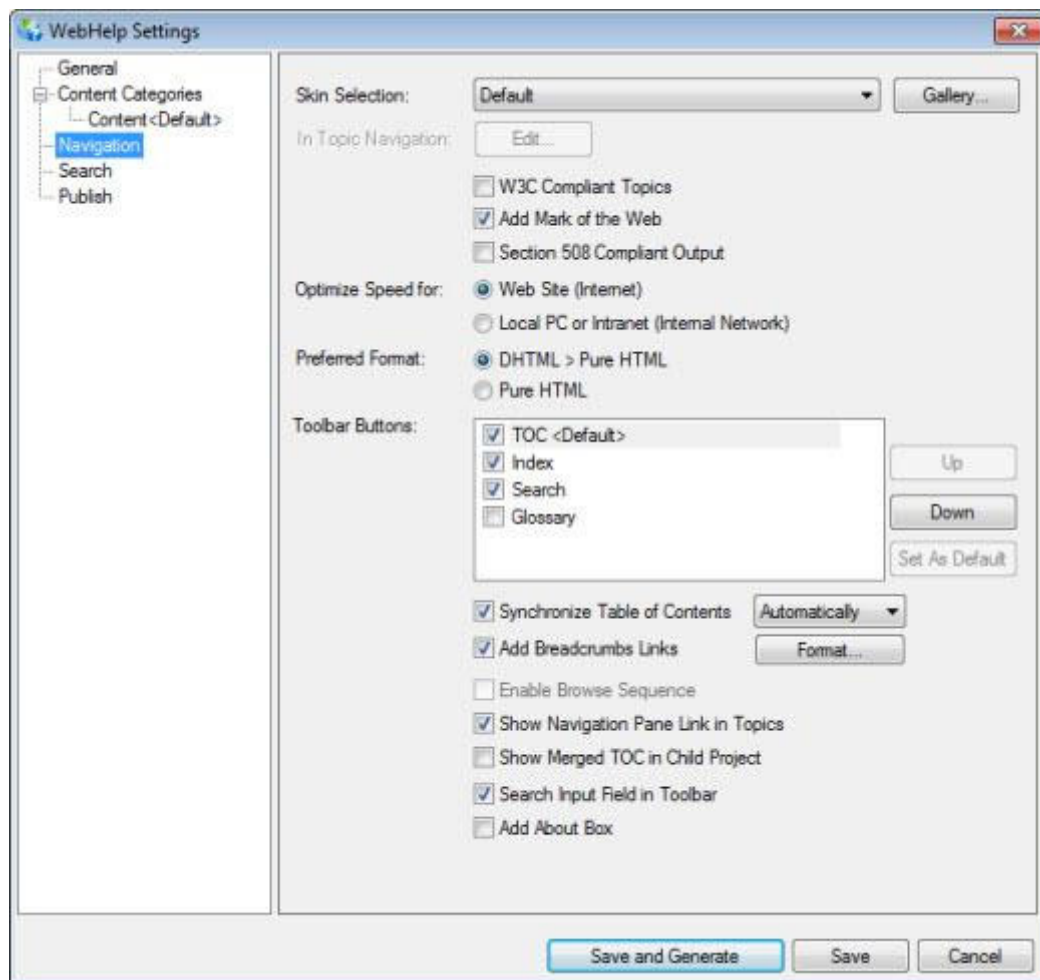| Field	Information
Content Title	Unless you are producing DUCC, you may as well leave this unchanged. DUCC within a merged setup is complex and you should not attempt it until you have first got your mind around how merged help works ordinarily.
Table of Contents / Index / Glossary	RoboHelp 7 introduced the ability to have more than one TOC, Index and Glossary. Select the required one here.
Variable Set	Variables were introduced in RoboHelp 6. Edit the values in the set you will use and then select that set here.
Default Topic	Select as required.
Language	Select as required. This will be the default language and will apply to content except where topics and / or paragraphs with a different

	setting.
Encoding	Leave the default unless your web administrator advises you that other encoding is required.
Map Files	Select as required.
Browse Sequences	Select as required.
Conditional Build Expression	The Conditional Build Expression option will apply only to the project being generated, not to other projects. That is desirable.
Apply to All Topics	Apply to all Topics was introduced in RoboHelp 8 and lets you apply a Master Page and / or CSS (Cascading Style Sheet) that will override whatever is set in your topics. I have not tested this with merged WebHelp but as long as you apply the same Master Page and / or CSS, I see no reason why it should not work.
Exclude Unreferenced Topics from Output	This was introduced in RoboHelp 9 and excludes any topic that is not in the Table of Contents or linked to from a TOC topic. The linking cascades so if the TOC has a link to a topic and that topic has a link to a topic not in the TOC, the second topic will be included, similarly if that topic has a link to another topic, the third topic will be included. If you use this option, check your output carefully to ensure the content is limited to what you want.
Exclude Baggage Files From Search	Certain types of baggage file will be included in search results unless you select this option.

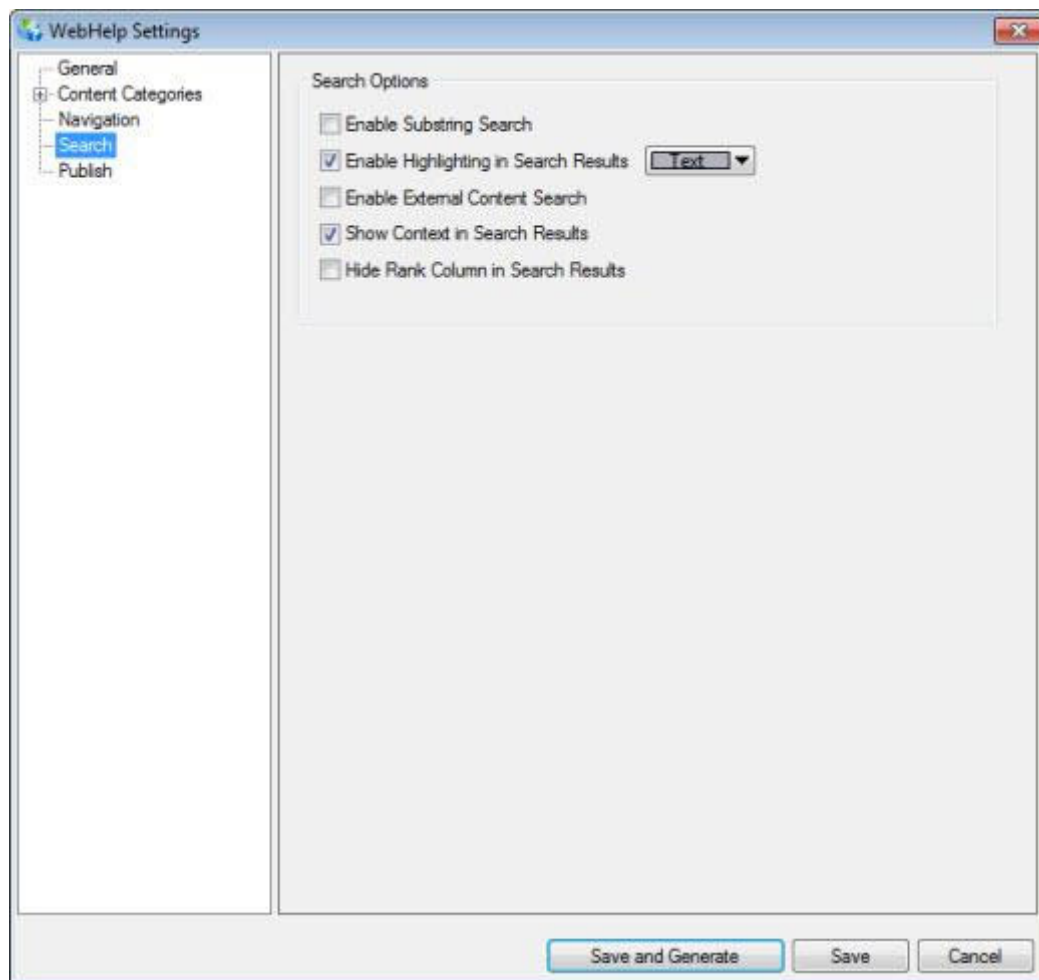## Navigation Settings

Figure 12.

Field	Information
Skin Selection	The skin for the Parent will apply to projects within the merge unless you open the child independently of the merge. Then the skin will be whatever you selected when generating that project.
In Topic Navigation	This field is only enabled if in Skin Selection you select "Traditional Style - No Skin". It provides options to include/exclude "Show/Hide" buttons for Navigation pane in a Topic. It also gives option to include/exclude browse sequence in topic.
W3C Compliant Topics	Tick if this compliance is required.
Add Mark of the Web	Use the Add Mark of the Web option in Additional Options while you are working locally, but note that links to PDF filesWord documents and some other files may not work locally.
Section 508 Compliant	Tick if this compliance is required.
Optimise Speed For Internet / Internal	Logically you should set the appropriate option. Often problems posted on the RoboHelp forums have been resolved by changing this setting to Local PC or Intranet.
Preferred Format	Select DHMTL unless you have a specific reason to select HTML. The latter will give you an index with A - Z characters rather than the normal index.
Toolbar Buttons	If you add any new toolbar buttons to the skin, make sure you select them when you set up this page. By default, they will be

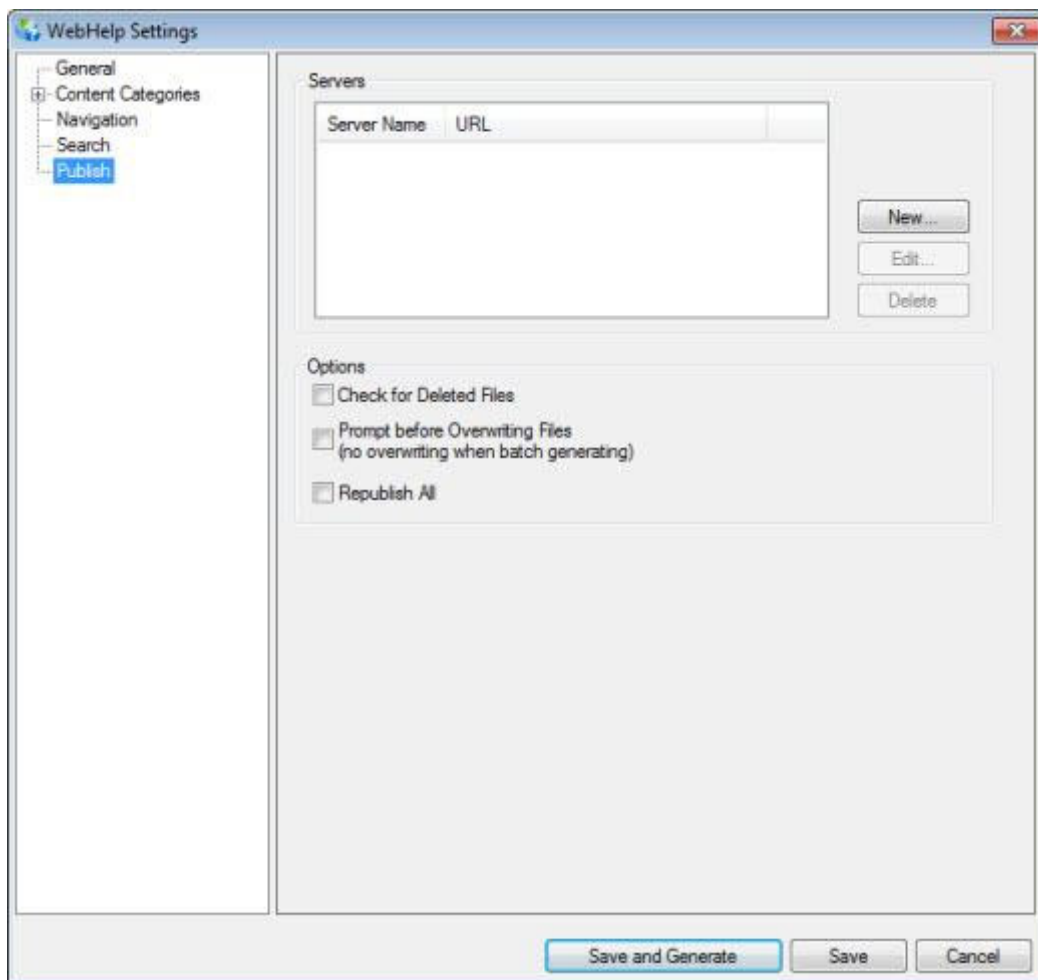|	listed, but not checked.
Synchronize TOC	This will work across the merge as long as you set the option in all the projects. It has been found that TOC synchronization works better if Optimise Speed (above) is set to Intranet.
Add Breadcrumb Links	In addition to any breadcrumbs you have included in a master page, this will add breadcrumbs to all topics.
Enable Browse Sequence	For some reason the browse sequences you selected in Content settings unless you also select this field.
Show Navigation Pane Link...	The Show Navigation Pane Link in Topics option is what causes the default text of Show to appear if you go to Windows Explorer and double-click a topic. It will also appear if your developers call the help incorrectly for merged WebHelp. See Calling Help for more information on that.
Show Merged TOC in Child Project	The Show Merged TOC in Child Project option is new in RoboHelp 8. Previously if you opened a child topic other than through the parent, you would only see the Table of Contents for that child project. By selecting this option, users will see the full merged TOC.
Search Input Field in Toolbar	Tick if this compliance is required.
Add About Box	For a long time people have asked how to remove the RoboHelp logo. Now you have to add it in deliberately. You can also change the image in the skin.

Search Settings

Figure 13.

The Search settings are the same as for any project and are not described here.

Publish Settings

Figure 14.

- Note that the Check for Deleted Files option in the wizard does not delete topics from the server; it simply checks for files that have not changed, but are missing from the server.

- It is recommended that you leave Republish all unselected. Then only updated topics and internal files will be published to the server thereby reducing the time taken to publish.

Donations

If you find the information and tutorials on my site save you time figuring it out for yourself and help improve what you produce, please consider making a small donation.

Topic Revisions