Links to Customer Files / Intranets

What's covered?

This page describes a method of creating links in your source files that will not be seen by users of the default help but will be seen if a javascript file in the output is edited. This enables customers to "create" links where required for their users. Typically these will be links to internal documentation or an intranet describing company specific procedures.

The method is available for both webhelp and compiled chm help, albeit using different scripts and procedures. It has not been tested in HTML5 help.

Do consider the workload involved for you and your customers before implementing this method.

- Introduction
- Demo
- What You Do
- What the Customer Does
- Downloads

Introduction

The costs associated with customising help topics explained in Customising Help will deter many customers and they will then seek an alternative. Often the request is along the lines of requesting some links to their own Word documents, PDF files, an intranet and so on.

Previously this has required some fairly clunky solutions but with the aid of some javascript, a rather more elegant solution is now available. Many thanks to my colleague Tony Harris for his work in producing this script which has some very neat options and to Pete Lees for making it work with compiled .chm help.

In the creation of every topic in the standard help, a piece of javascript is inserted (there are ways of adding it retrospectively to existing help topics). What this does each time a topic is opened is:

- Check to see if the javascript has been amended.
- If it has not, then no hyperlink will be displayed.
- If the customer has entered a path and filename, the link will be displayed with either the default text or the customer's amendment or an image.
- The target can be displayed within the help or in a new window.

Demo

Click these links to see it work.

Demo Topic 1 - Shows a topic where nothing has been defined in the javascript file.

Demo Topic 2 - Shows a link with the default text.

Demo Topic 3 - Shows a link with alternative text.

Demo Topic 4 - Shows an image that links to a website in a new window.

May 2017 - A recent check indicates that because of changes to browser security, the image in

> Demo Topic 4 will not display if the page is viewed locally rather than being on a web server.

These two illustrations show the same topic, firstly as it was supplied to the customer and then after they had amended the script as described in the next section. Exactly what you want, the link is displayed only in topics where it is required. It can point to additional or alternative procedures.

The beauty of this method is that the script is installed once in the source files and the work is then done by the customer who has control over the cost.

What You Do

This section describes the set up for the technical author. What the customer has to do with the installed help files is covered in the next section. Whilst these instructions were written for RoboHelp users, they can be used with any HTML editor.

Inserting the script in the topic

You need to decide upon a standard place to insert the script. My suggestion is that it goes immediately below the heading for the topic. Customers using the idea can then train their staff that they click this link instead of the help or as well as, it depends on whether they are providing alternative help or supplementary help.

Please note that the scripts and procedures for webhelp and compiled chm help are different.

Webhelp

If you are producing webhelp:

1. Save the javascript files in the root of your project. See the Downloads section at the end of this topic.

2. Click inside this box and press Ctrl + C

```
<script type="text/javascript"
language=JavaScript1.2
src="custom_link.js"></script>
```

3. Paste this script into your topic between the head tags <head> and </head>. If the topic is at a different folder level to the custom_link.js file, you will need to amend it to show the relative path. Click here for details.

4. Click inside the box below and Ctrl + C

```
<script type="text/javascript"
language=JavaScript1.2
src="custom_linklist.js"></script>
```

5. Paste this text into your topic where you want the link to be displayed. Again you will need to modify the relative path.

6. Save the file and make sure the two javascript files now appear in your baggage files.

In the RoboHelp WYSIWYG editor you will see a red square where the script has been inserted. If you click on Preview, you will get an error message which can be ignored and you will then see the topic. The red square will not be displayed but the link can only be seen at the output stage.

Compiled CHM Help

If you are producing compiled chm help:

1. Save the javascript files in the root of your project. See the Downloads section at the end of this topic.

2. Click inside this box and press Ctrl + C

```
<script type="text/javascript"
src="ms-its:yourhelpfile.chm::/custom_link_chm.js">
</script>
```

3. Paste this script into your topic between the head tags <head> and </head>. The same script is used in all topics regardless of the level of the folder. Note that the chm filename needs to be amended from yourhelpfile.chm to whatever name you will give to your file.

4. Click inside the box below and Ctrl + C

```
<script type="text/javascript">GetExternalScript()
</script>
```

5. Paste this text into your topic where you want the link to be displayed.

6. Save the file and make sure the two javascript files now appear in your baggage files.

In the RoboHelp WYSIWYG editor you will see a red square where the script has been inserted. If you click on Preview, you will get an error message which can be ignored and you will then see the topic. The red square will not be displayed but the link can only be seen at the output stage.

The Script Explained

The logic behind the script is that if nothing is done with the custom_linklist.js or the custom_linklist_chm.js files in the output, then no links will be displayed.

If however, the required lines are added to the custom_linklist.js or the custom_linklist_chm.js files in the output, then the link will be displayed in accordance with the settings. This is described in more detail in What the Customer Does.

Bulk Changes

If your help has not yet been written, You could add the lines to a template so that it automatically gets added to every topic.

If the help has been written, consider using a multi file find and replace tool to paste the script into the same location in every source file.

If your topics were based on a template, you could add the script to the header or footer and then choose to apply it to all topics based on that template. You would need to consider whether that would impact on any other header and footer information in a topic based on that template.

However you do it, bear in mind that for webhelp you need to amend the relative paths if you have topics at different folders levels (the relative path to the javascript files will vary for each level).

Generating the Help

Just generate your help in the normal way. No links will be seen in the initial output and nothing will appear in any printed documentation you create from the source files. As the links are customer specific, this is the desired result anyway.

If you have to produce both webhelp and compiled chm help, you will need to follow both procedures above and use conditional text.

What the Customer Does

Using a text editor, the customer opens the custom_linklist.js or custom_linklist_chm.js file as appropriate in the output. The custom_link.js and custom_link_chm.js files require no editing.

The act of generating webhelp will place a blank custom_linklist.js file in the output. If you are generating compiled chm help then you will need to manually add a blank custom_linklist_chm.js file to the same location as the chm file.

The instructions have been written into the custom_linklist.js and custom_linklist_chm.js files. The screenshot below shows the custom_linklist.js file as it is supplied in the webhelp output. Note the empty lines above the comments. The detail of the instructions in the custom_linklist_chm.js file is different to the webhelp version shown below.

```
// To edit this file add lines ABOVE in the format
//
// TO OPEN IN THE HELP WINDOW
// addLink("","",null,null);
// For example, addLink("demo.htm","procedure.pdf",null,null);

// TOP OPEN IN NEW WINDOW
// addLinkExternal("","",null,null);
//// For example, addLink("demo.htm","http://www.website.xxx",null,null);
//
// The instructions for completing either type of line are the same.
//
// FIRST PARAMETER
// In the first set of quotes, enter the name of the file the link is to appear in.
// If the file name is unique, you need only enter the file name.
// If the same file name appears in more than one folder, it is necessary to
// uniquely identify it.
// To uniquely identify the file, follow the indented instructions.
//     1. All paths must start with a single forward slash.
//     2. It is not necessary to enter the full path although although you can do
//        so if preferred.
//     3. All that is mandatory is sufficient to uniquely identify the file.
//     EXAMPLES
//     There are three files called demo.htm
//     File one is in folder /webhelp
//     File two is in folder /webhelp/demo_02
//     File three is in folder /webhelp/demo_02/demo_03
//     You can enter the full path
//     /webhelp/demo.htm
//     /webhelp/demo_02/demo.htm
//     /webhelp/demo_02/demo_03/demo.htm
//     But the following is sufficient
//     /webhelp/demo.htm
//     /demo_02/demo.htm
//     /demo_02/demo_03/demo.htm
//// SECOND PARAMETER
// In the second set of quotes, enter the target for the link, this can be any file, an i
```

The procedures in rest of this topic relate are written specifically for webhelp. You will need to slightly modify them for compiled chm help.

Single Link in a File

First let's look at adding a single link in one file. The customer simply creates a new line above

the comments using either of the options:

- addLink - Opens the target in the help window.
- addLinkExternal - Opens the target in a new window.

To start with the links will look like this

addLink("","",null,null);

OR

addLinkExternal("","",null,null);

Both types of link are completed in the same way.

1. In the first set of quotes, they enter the name of the file in which the link is to appear. Provided the custom_linklist.js file does not contain any duplicated file names, then only the file name need be entered. Note the instructions in the js file if any filenames are duplicated.

2. In the second set of quotes, they enter the target for the link, this can be any file, an intranet or a web site. Unless the target file is in the same folder as the javascript, the relative path will be required using forward slashes. For intranets and websites, use the format http://www.website.xxx.

3. The first null is an option. If the customer prefers to show an image instead of text, then they enter the filename of the image. It is important that the filename is added within quotes.
Unless the image is in the same folder as the javascript, the relative path will be required using forward slashes. Note the requirement to enter alternative text.

4. The second null is an option. If the customer does not want the default text for links and does not want to use an image, then alternative link text can be entered that will only apply to this link. The text must be added within quotes and can include style tags (such as <p>).

5. The default text is the last line of the file. If the customer prefers something other than Click here for internal procedures, this is where it should be edited.

6. You will notice that I have applied a class to the <p> tag. Because of the way the javascript file writes the link, it would otherwise display underlined but with the text formatted as for your normal <p> tag. To make it display as a link, you will need to create a class in your style sheet. You can see this in the demo project that can be downloaded below.

An example line might be

addLink("demo.htm","procedure.pdf",null,null);

OR

addLinkExternal("demo.htm","http://www.website.xxx",null,null);

Click here for more more examples.

Multiple Links in a File

Although the script <script language=JavaScript src="custom_linklist.js"></script> can only be inserted once in any topic, it can display more than one link at that location.

So if the customer want two procedures shown, say one for North Region and one for South Region, two lines would be created something like this:

addLink("demo.htm","north.pdf",null,"<p class=Custom>Click here for North Region instructions.</p>");

addLink("demo.htm","south.pdf",null,"<p class=Custom>Click here for South Region instructions.</p>");

In the output, it would look something like this.

If you particularly need links in more than one location in the file there are two options:

- Name the existing files custom_link_01.js and custom_linklist_02.js and then insert additional files called custom_link_02.js and custom_linklist02.js. I have not tested this method yet but I think it should work.
- Use the earlier method but please note the drawbacks. Click here for more information.

It is vital the customer is aware that they must keep a copy of the custom_linklist.js or custom_linklist_chm.js file with all their addLink rows so that it can be restored if the help is overwritten at any time, for example when the software is upgraded.

If the customers place any target files within the help structure, they must ensure that they are also backed up.

Downloads

Download 1 - Webhelp

Click here to download a mini project demonstrating how things are set up and showing the links as the customer will see them, or not!

To run the demo, extract the files using WinZip's extract function to preserve the structure. Then go to \!SSL!\WebHelp and run the index.htm file.

To alter what is displayed, you need only edit the custom_linklist.js file in the output. If you want to generate your own output, I suggest you create it in a new folder so as to retain the supplied output.

For the javascript files that you will require in your project, use the custom_link.js and custom_linklist.js files that are in the root of the demo project, not those in the demo output.

Download 2 - Compiled chm help

Click here to download a mini project demonstrating how things are set up and showing the links as the customer will see them, or not!

To run the demo, extract the files using WinZip's extract function to preserve the structure. Then go to \!SSL!\Microsoft_HTML_Helpand run the helpfile.

To alter what is displayed, you need only edit the custom_linklist_chm.js file. If you want to generate your own output, I suggest you create it in a new folder so as to retain the supplied output.

For the javascript files that you will require in your project, use the custom_link_chm.js and custom_linklist_chm.js files that are in the root of the demo project, do not use the custom_lnklist_chm.js file in the demo output.

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

Revisions