

Developing dual-use content

**Using FrameMaker to Deliver
Customized Print and Online Content from a Single Set of Files**

Scriptorium Publishing Services, Inc.

P.O. Box 12761, Research Triangle Park, NC 27709

919-481-2701 or sales@scriptorium.com

<http://www.scriptorium.com>



This white paper discusses how to take advantage of FrameMaker features to deliver customized print and online content from the same set of files. This process requires a thorough understanding of FrameMaker features, including cross-references, conditional text, and character tags. These features let you develop a single set of FrameMaker files from which you can generate printed books, Acrobat files, and online help files. Although all the information is stored in a single set of files, the content is displayed differently in each medium.

In addition to FrameMaker knowledge, you'll need knowledge of a third-party tool, either WebWorks Publisher or MIF2RTF, to create the output. This paper focuses on using WebWorks Publisher.

Cross-references

You can use two templates to customize the appearance of your cross-references for print and online use. To do so, you set up a cross-reference format in the print template and then modify that format in the online template. When you import formats, the cross-reference format is updated, which means that the appearance of every cross-reference with that particular format changes.

For example, an appropriate format for your print document might be the following:

```
"<$paratext>" on page <$pagenum>
```

This format would result in a cross-reference that looks like this:

```
"Cross-references" on page 2
```

In your Adobe Acrobat (PDF) version, you might want to color-code and underline the cross-reference to ensure that readers notice that it is a live link.¹ To accomplish this, you create a character format that applies the color and the underline, and include this format in your cross-reference definition. I usually name my character tag "blue." So, your new cross-reference format looks like this:

```
<blue>"<$paratext>" on page <$pagenum><Default Para Font>
```

The <Default Para Font> command at the end of the cross-reference sets the text back to the paragraph default, which ensures that the blue doesn't "bleed" into the text following the cross-reference.

Another technique is to put in two cross-references (each with a different format) and tag them with conditional text so one is shown in print and one in the online version. I consider the single cross-reference with two templates a more elegant solution.

1. Remember, FrameMaker cross-reference become live hyperlinks when you generate Acrobat data while converting the Frame files to PDF.

The blue character tag looks like Figure 1.

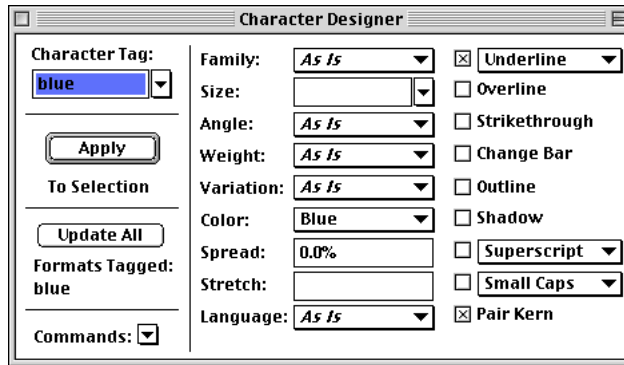


Figure 1: Character tag “blue” in the PDF template

In the print version, set up the character format “blue” so that it does not apply the blue color or the underline. Simply change those attributes by setting them to As Is (see Figure 2).

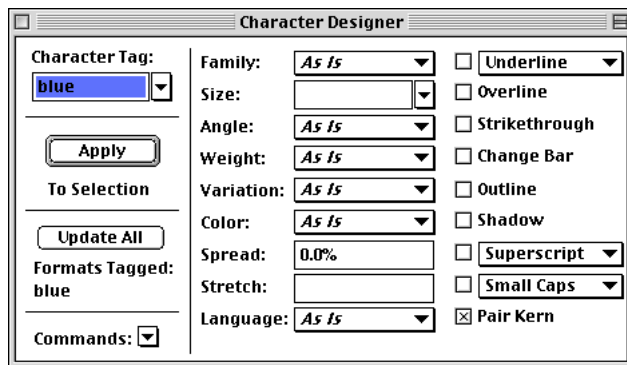


Figure 2: Character tag “blue” in the print template

For HTML or WinHelp output, you may want to modify the cross-reference format again. Typically, you do not need the page number to be included in these formats, and the links are automatically colored and underlined by the browser or by the WinHelp engine. As a result, you want a cross-reference format like this:

<\$paratext>

This displays the content of the referenced paragraph without any special formatting, quotation marks, or page number. Table 1 summarizes the settings needed in the templates.

Table 1: Template settings for customized cross-references

	Cross-reference format	Character tag “blue”
Print (option 1)	<blue>“<\$paratext>” on page <\$pagenum><Default Para Font>	All settings are As Is. Does not change appearance of text.
Print (option 2)	“<\$paratext>” on page <\$pagenum>	N/A
PDF	<blue>“<\$paratext>” on page <\$pagenum><Default Para Font>	Changes text color to blue and adds an underline.
Online help	<\$paratext>	N/A

Conditional text

FrameMaker’s conditional text feature lets you label text and graphics in your files for inclusion only in certain versions of your document. WebWorks Publisher also offers a feature called *pass-through code*. This lets you use a conditional text tag to identify data that you want to place directly into your output files without processing. This is often used to place JavaScript code directly into your output.

Restricting content to a single medium

Often, you need conditional text for transition material that is useful in the printed version of a document but that does not work in the online help. For example, at the end of a chapter, you might include a summary section that highlights the most important parts of the just-completed chapter and that discusses how this information relates to the next chapter. But if you plan to convert your document to online help, these transition section should be removed because you have no way of knowing whether an individual will read them before they move on to the next chapter.

Another place where conditional text might be required is if you use statements such as, “as discussed in the previous section.” In online help, the reader would have no way of accessing the previous section. Instead, the online help needs a specific cross-reference—something like, “as discussed in [Understanding the Windows Interface](#).” The reader could then click on the link to read the referenced material.

You need at least two conditional text tags: a print-only tag and a help-only tag. With those two tags, you can include all the information for both print and online in your document and then tag the media-specific information with the needed conditional text tag.

For example, at the beginning of a chapter, your print version might read like this:

This chapter describes how to organize your closets. You'll learn how to identify items that you want to throw out, what hardware is available to use in your closet, and how to put it all together for a more organized, useful closet. Read on for all the details!

But if this file were online help, you would want something like this:

```
Organizing your closets
  Identifying items to throw out
  Hardware for the closet
  Putting it all together
```

Each of these items would be a cross-reference to the appropriate section.

Your final document would look like Figure 3.

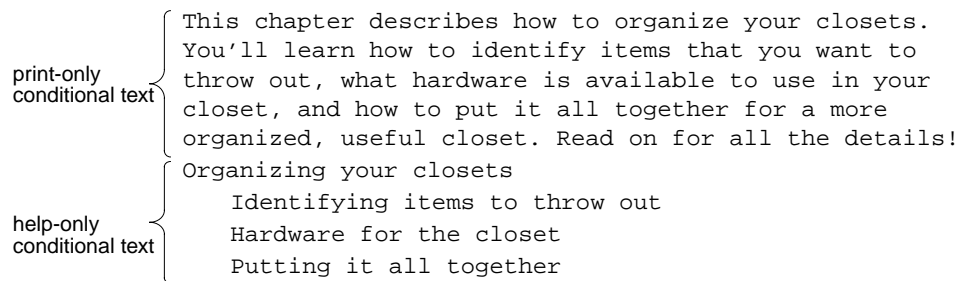


Figure 3: A FrameMaker document with conditional text tags

Inserting code into output files

Another useful conditional text trick is inserting pass-through code. If you are using WebWorks Publisher to create your online files, you can use a special conditional text tag to flag data that you want to send to the output files without processing. You could, for example, use this technique to embed HTML code, as shown in Figure 4.

```
When describing a page range, use an en dash.
<!--This is a comment tag-->
Use an em dash to indicate a break in thought.
```

Figure 4: Embedded HTML code in FrameMaker is tagged with conditional text

In this example, the second line of text would have a conditional text tag applied. You could create a tag, for example, called HTML. You would then need to set up your WebWorks Publisher conversion to pass through the HTML conditional text tag without any processing, as shown in Figure 5.

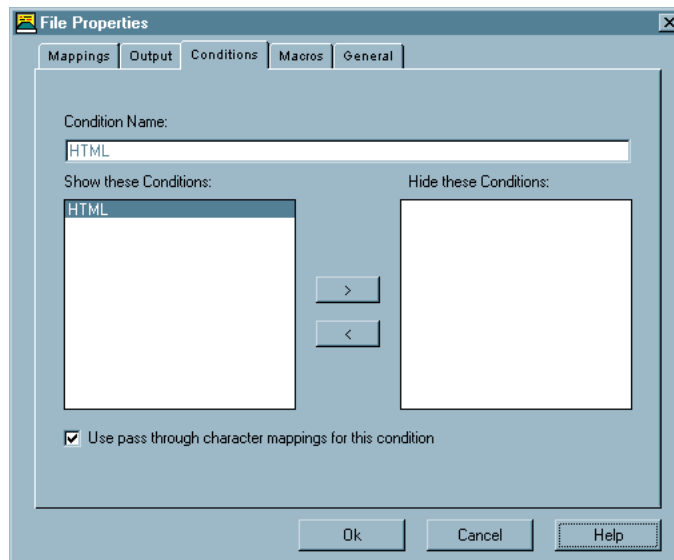


Figure 5: Setting up WebWorks Publisher to pass through HTML conditional text

The end result, in HTML, would look something like this:

```
<P>When describing a page range, use an en dash.</P>  
<!--This is a comment tag-->  
<P>Use an em dash to indicate a break in thought.</P>
```

Figure 6: Generated HTML includes the code with no paragraph tags added

Markers

FrameMaker's markers let you insert hidden text into your documents. WebWorks Publisher can process this text in many different ways.

Processing URLs

Markers are often used to insert URLs into a document. In the print version, you want to display the actual URL (such as `http://www.scriptorium.com`), but in the online version, you want that URL to become a live hyperlink. You can accomplish this by defining custom markers.

First, you need to create two custom markers types, StartURL and EndURL.

Then, you insert a StartURL marker with the text of the marker at the beginning of the link and insert EndURL with no content¹ at the end of the link, as shown in Figure 7.

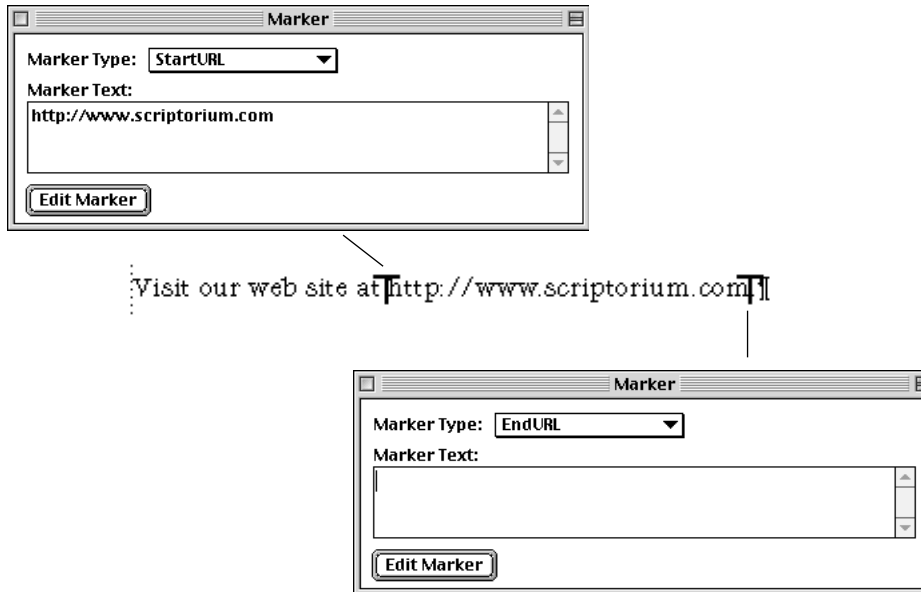


Figure 7: Using custom markers to insert URLs

In WebWorks Publisher, you need to map these custom marker types. The first marker type, StartURL, should create this result:

```
<A HREF="$DATA;" >
```

The EndURL marker type should create this output:

```
</A>
```

1. The EndURL marker is used to close the hyperlink in HTML.

Map the StartURL marker type to GotolinkFileandTag and the EndURL marker type to CloseAnchor (see Figure 8).

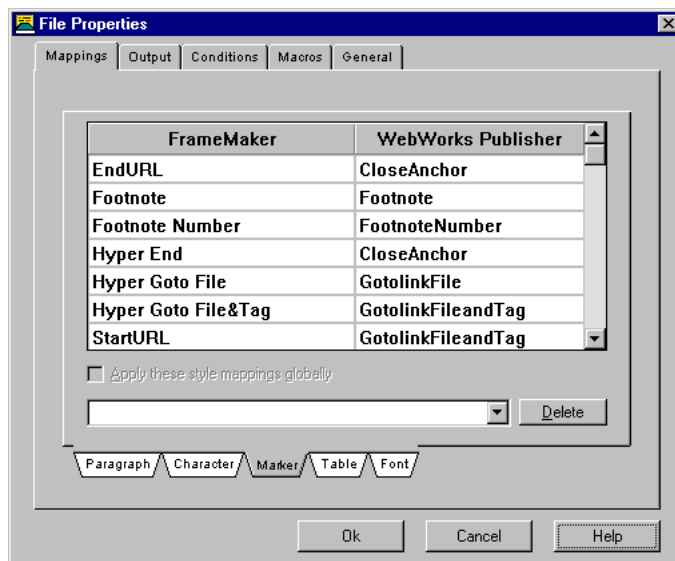


Figure 8: Mapping marker types

The result, in the output HTML, looks like this when you convert the example sentence above:

```
<P>Visit our web site at <A HREF="http://www.scriptorium.com">http://www.scriptorium.com</a>.
```

Creating arbitrary links

For any links you need within the files in your WebWorks Publisher project, you can use cross-references. But in some cases, you need to create a link to, for example, another help file. You can use markers to build a link to any topic in any help file.

You need the name of the topic and the help file, and you need to create a “live” area by tagging the live text with a new character tag called hotspot. The hotspot character style does not change the appearance of your text (because you do not want any changes to be displayed in the printed version of your books). However, hotspot lets you identify the area that will become the link.

At the end of the hotspot text, insert a marker with a custom type (for example, CustomHotspot). In the marker, type the text of the link, in the format *topic@helpfile*; Figure 9 shows an example.

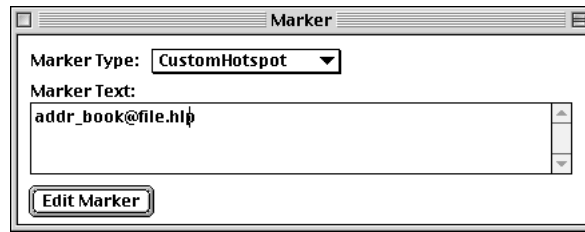


Figure 9: Inserting the link destination into a custom marker

In WebWorks Publisher, define your hotspot character tag to start a link, like this:

```
{\\uldb $DATA;}
```

The CustomHotspot marker provides the link destination:

```
{\\v $DATA;}
```

Who should do this?

Setting up FrameMaker files with dual-purpose content can be a tedious and time-consuming exercise. It requires a high level of FrameMaker knowledge and close attention to detail. Is this a worthwhile investment?

The answer depends on your environment. If you have a large amount of information in FrameMaker that needs to be delivered both on paper (or PDF) and in online help or HTML, this solution is the right choice. The time spent configuring the FrameMaker files is saved later because maintenance time is reduced. The alternative to developing online help content in FrameMaker is to export the FrameMaker content to RTF, import it into RoboHelp/Microsoft Word, and then rework the files to recreate links, correct formatting lost during the conversion, and reinsert graphics. The FrameMaker alternative is much faster, and it has the added advantage of eliminating duplication of your content. Instead of maintaining a copy in RoboHelp (for help) and a copy in FrameMaker (for print), you have a single set of source files. This alone makes the FrameMaker-only approach very attractive.

Contacting us

Scriptorium Publishing Services, Inc. excels at transforming complex technical ideas into clear, concise documents. Our clients, who range from start-ups to Fortune 500 companies, rely on us for the full spectrum of technical publishing services—everything from turnkey documentation to specialized technical editing and consulting. Our expert, talented staff thrives on working with challenging new technology in a deadline-driven environment.

If you have any questions about Scriptorium Publishing Services, Inc., contact:

Scriptorium Publishing Services, Inc.

P.O. Box 12761

Research Triangle Park, NC 27709

919-481-2701

sales@scriptorium.com

<http://www.scriptorium.com>