

# 9

## *Using Third-Party Help-Authoring Tools*

Using an HTML document-authoring tool facilitates the creation of HelpSets. Such third-party tools automate many of the JavaHelp development tasks that would otherwise be tedious and time-consuming. To help you make decisions about using third-party authoring tools, this chapter covers the following topics:

- How JavaHelp relates to third-party tools
- How third-party tools work

### *How JavaHelp Relates to Third-Party Tools*

Most third-party help-authoring tools enable you to achieve the cherished goal of “write once, run anywhere.” That is, with a single authoring effort, you can create an online help system to be deployed as JavaHelp, WinHelp, or HTML Help.

A good third-party tool attempts to hide the administrative details of JavaHelp. It may even appear that you don’t even need to understand the workings of the help system, as detailed in the preceding chapters. But a good understanding of JavaHelp aids in diagnosing problems and limitations in the authoring environment. It enables you to go “behind the scenes” to correct or work around such situations. And if you ever need to abandon the third-party tool, you can continue maintaining the help system using the techniques discussed in this book.

### *How Third-Party Tools Work*

Sun Microsystems reports that many help-authoring tools offer or plan to offer JavaHelp support. You should approach the acquisition of a third-party help-

authoring tool in the same way you would consider purchasing any other software:

- Make sure the software offers the features you need and the ease-of-use you want.
- Make sure the price is right.
- Visit the tool vendors' web sites to get more information on their products. Some offer free trial versions to help you decide if the product is right for you.

Sun's JavaHelp site ([www.java.sun.com/products/javahelp](http://www.java.sun.com/products/javahelp)) lists companies offering JavaHelp third-party tools. Some of the popular products include ForeFront *ForeHelp*, Blue Sky *RoboHelp*, and Wextech *Doc-to-Help*. These products are available for the Windows operating systems; RoboHelp and Doc-to-Help require MS-Word. In the remaining sections in this chapter, I discuss how these tools can assist you with help authoring.

### ***ForeHelp***

*ForeHelp*, from ForeFront, is a standalone help-authoring tool. You don't need to use any other application—in particular, MS-Word—in conjunction with it. ForeHelp provides an integrated environment in which you manage projects, write topics, create HelpSet data and navigation files, and enhance the HelpSet.

Creating and setting up a JavaHelp project is fairly easy with ForeHelp. The main flaw, in my opinion, is that ForeHelp doesn't let you organize help topics into a meaningful subdirectory structure, as shown in Chapter 3, *Planning the JavaHelp Project*. Instead, ForeHelp places all topic files in one directory. As long as you stay within the ForeHelp authoring environment, this is fine, because ForeHelp manages all the data for you. But if you later decide to use the files without ForeHelp, organizing the topics files into subdirectories will be a manual task.

ForeHelp's help-topic editor, shown in Figure 9-1, is similar to MS-Word and other Windows word processors. You can quickly format text, justify paragraphs, and apply HTML tags.

The frame to the left of the topic editor shows all the project's help topics. To edit a particular topic, you simply select it from the list. This opens the topic "file" and loads it into the editor. (Actually, ForeHelp maintains all of a project's topics in a single database. It doesn't create separate HTML-format files until you "build" the project.)

As you create help topics, you don't have to worry about assigning map IDs or help topic filenames because ForeHelp automatically assigns them for you.

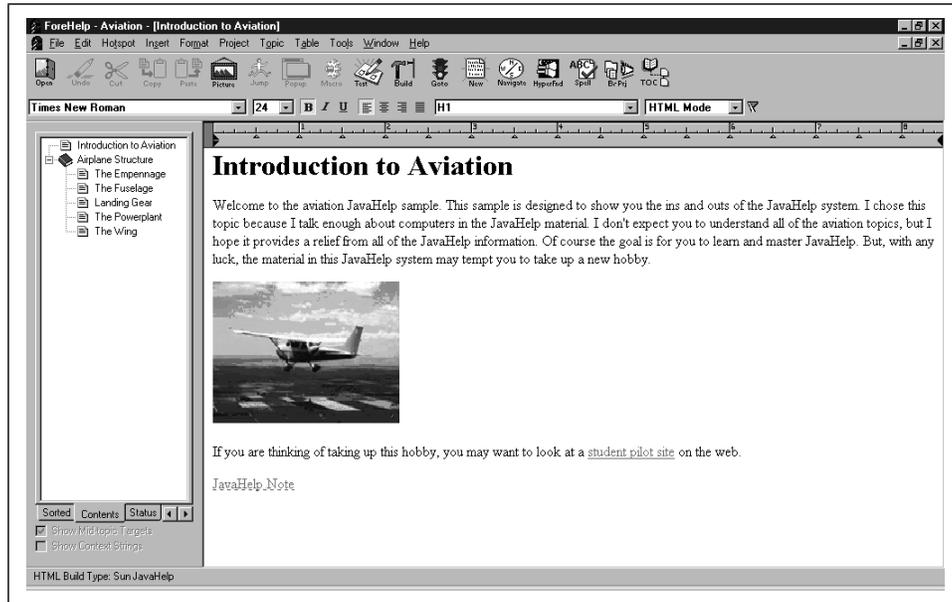


Figure 9-1. ForeHelp's topic editor

Recall from Chapter 5, *Creating HelpSet Data and Navigation Files*, that creating a JavaHelp TOC file was a detailed manual task. ForeHelp makes this process much easier. The Contents Editor, shown in Figure 9-2, provides a simple and convenient interface for organizing your topics into a hierarchy. (In the ForeHelp model, a category is represented by a book icon; JavaHelp uses a folder icon.) ForeHelp makes it particularly easy to keep track of which topics you've included in the TOC.

ForeHelp also provides a convenient interface for creating a project's index. You use a keyword editor to specify index items for each help topic. When you build the help system, ForeHelp creates an alphabetized index file automatically.

Creating the word search index is the easiest of all: ForeHelp generates it automatically when you build your JavaHelp project.

In Chapter 6, *Enhancing the HelpSet*, I showed you different ways to enhance your HelpSet. ForeHelp provides features for making these enhancements easy. For example, you can specify that a topic should appear as a pop-up or in a secondary window.

ForeHelp provides a great feature for specifying related topics within each of your help topics. You use a wizard to set up relationships between topics, as shown in Figure 9-3.

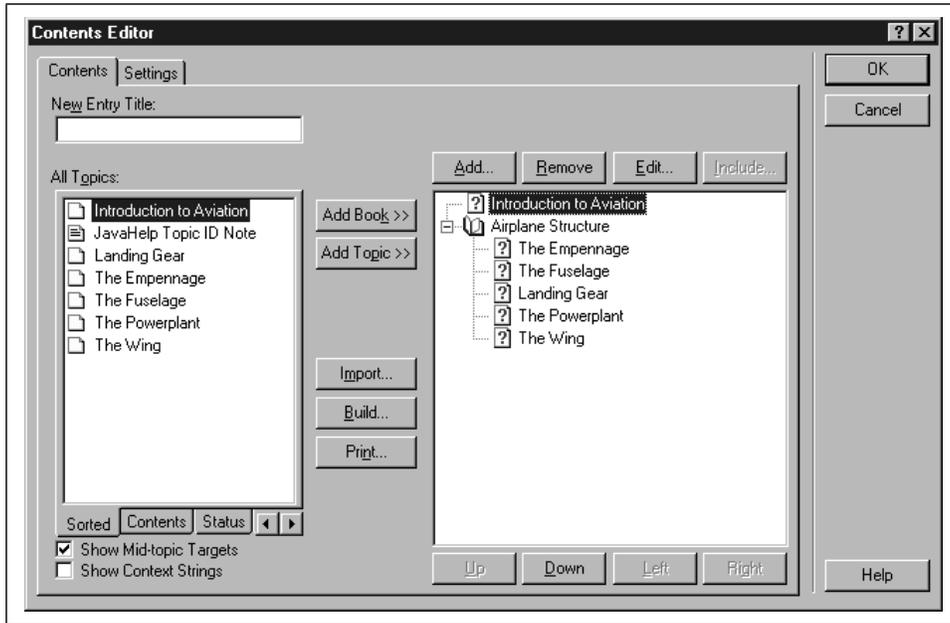


Figure 9-2. ForeHelp's contents editor

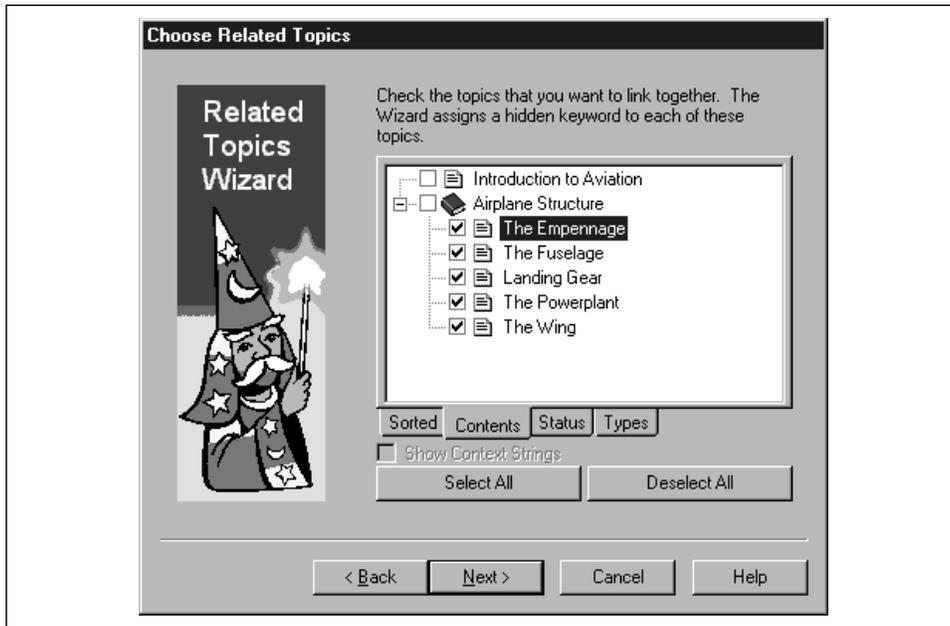


Figure 9-3. ForeHelp's related topics wizard

When the help system is running, a button or text (depending on which you choose during development) appears in each of the related topics. When the user clicks the button or text, the help system displays a related topics window, as shown in Figure 9-4.

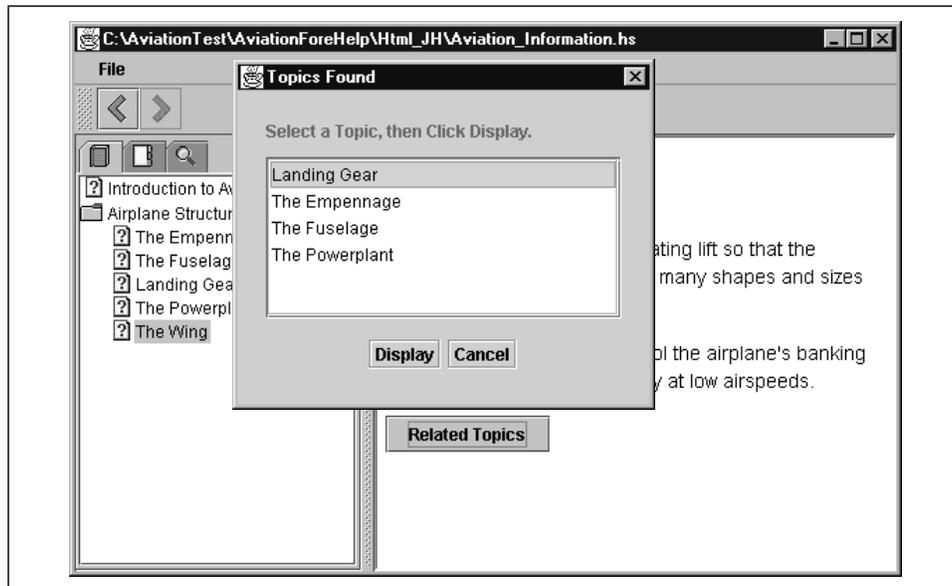


Figure 9-4. ForeHelp's related topics selection dialog

When the user selects a topic from this window, the topic appears in the content pane.

Finally, ForeHelp offers the JavaHelp Settings window, shown in Figure 9-5, for specifying JavaHelp navigation alternatives such as including or excluding navigation components, adding new navigation tabs, and editing existing navigation tabs.

You can also use this window to merge HelpSets and to use and specify a JAR file. If you choose to use a JAR file, ForeHelp automatically compresses and encapsulates the HelpSet files into a JAR when you invoke the compile command.

You can get more information on ForeHelp and download a free preview release of ForeFront software from their web site, at <http://www.ff.com>.

## ***Doc-to-Help and RoboHelp***

WexTech *Doc-to-Help* and Blue Sky *RoboHelp* both depend on MS-Word being installed on your computer. This dependency could be a main reason for not using those tools: extra expense. On the other hand, authors who already use MS-

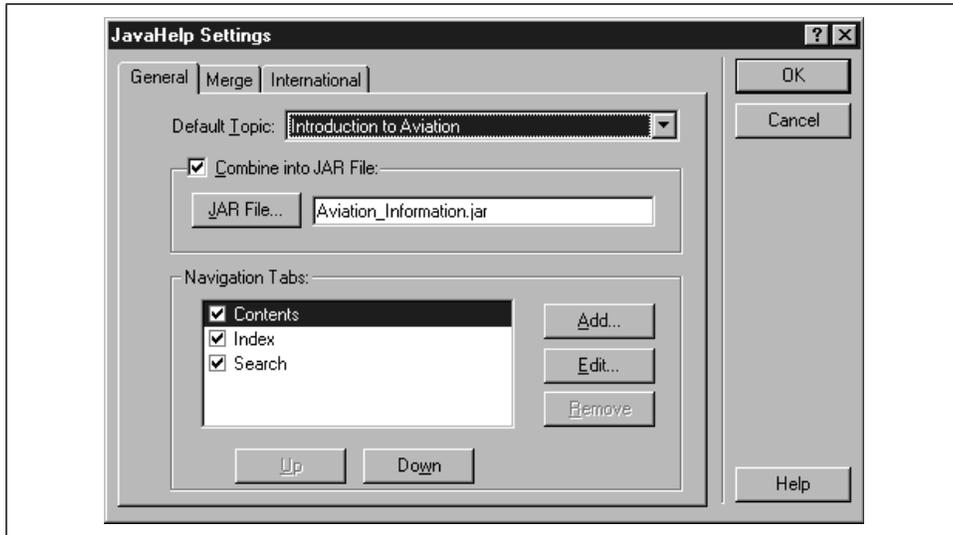


Figure 9-5. ForeHelp's JavaHelp settings window

Word may adapt to these tools more quickly, given their familiarity with the MS-Word user interface.

The integration with MS-Word offers the ability to create both online help and hardcopy documents from the same source. While this feature enables you to save time by creating two types of documentation at the same time, many professional writers, including me, argue that online help and hardcopy documents should be created separately. Because of the way users use online help versus hardcopy documents, you shouldn't use the same format and style for both types of documentation.

Like ForeHelp, the Word-dependent tools provide a friendly graphical interface to assist you with managing projects, writing topics, creating HelpSet data and navigation files, and enhancing a HelpSet. Writing the help topics is somewhat different, since you write the topics in MS-Word itself. The tools offer functions to extend Word's capability so that you can manage your online help project.

Most of the functions in the Word-dependent tools are similar to the functions in ForeHelp; they all use graphical components to make the help development process easy.

You can get more information on these third-party tools from the companies' web sites. At the WexTech web site, <http://www.wextech.com>, you can download a free trial version of Doc-to-Help. Blue Sky's web site is at <http://www.blue-sky.com>.