

Documentation Procedures



Doc Procedures, Version X

Copyright

©1999-2005 Wise Solutions, Inc. All Rights Reserved.

This documentation and the accompanying software are copyrighted materials. Making unauthorized copies is prohibited by law. No part of the software or documentation may be reproduced, transmitted, transcribed, stored in a retrieval system or translated into any human or computer language without prior written permission of Wise Solutions, Inc. Wise Solutions, Inc. asserts its "Moral Right" to be identified as the author of this work, in all jurisdictions which recognize the "Moral Right."

Notice

Unless otherwise provided by written agreement with Wise Solutions, Inc., this publication, and the software sold with this publication, are provided "as is" without warranty of any kind either express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. The entire risk arising out of the use or performance of this publication and software remains with you. In no event will Wise Solutions, Inc., or any of its suppliers, be liable for any lost profits, lost savings, direct, incidental or indirect damages or other economic or consequential damages, even if Wise Solutions, Inc., or its suppliers, have been advised of the possibility of such damages. Wise Solutions, Inc. reserves the right to modify this document at any time without obligation to notify anyone. In no event shall Wise Solutions, Inc.'s or its suppliers' liability under this agreement exceed the sum of any amounts paid hereunder by the customer to Wise or the supplier.

Trademarks

Wise Solutions, Inc. owns a number of registered and unregistered Trademarks and Service Marks (the "Marks"). These Marks are extremely valuable to Wise Solutions, Inc. and shall not be used by you, or any other person, without Wise Solutions, Inc.'s express written permission. The Marks include, but are not necessarily limited to the following: Application Isolation Wizard™; ApplicationWatch™; ConflictManager®; ExpressBuild™; Installation Development Life Cycle™; InstallBuilder®; InstallMaker®; InstallManager®; InstallTailor™; MSI Debugger™; MSI Script™; PackageManager™; Preflight Deployment™; SetupCapture®; SmartMonitor™; SmartPatch®; Software Distribution Made Easy™; Software Installations Made Easy®; Visual MSIDiff™; Virtual Capture™; WebDeploy™; Wise Installation System®; Wise Package Studio®; Wise Software Repository™; Wise Solutions®; WiseScript™; WiseScript Express™; WiseUpdate®; WiseUser®; and the Wise Solutions logo.

In addition to Wise Solutions, Inc.'s Marks, some Wise Products may include Trademarks or Service Marks owned by other corporations. These other Marks include, but are not necessarily limited to Microsoft® Windows® and Microsoft® Visual Studio® .NET, which are registered Trademarks of Microsoft Corporation.

You shall not use any of the Trademarks or Service Marks of Wise Solutions, Inc., Microsoft Corporation, or any other entity, without the express written permission of such Trademark or Service Mark owner.

Wise Solutions, Inc.

47911 Halyard Drive; Plymouth, Michigan 48170 USA
Phone: 734-456-2100 • Fax: 734-456-2456 • info@wise.com • www.wise.com

Contents

1	Standards for Writing Wise Documentation	8
	Focus and Audience	9
	Sentence Construction	10
	Chapter Openings	12
	Topic Introductions	13
	Topic Heading Guidelines	13
	Use of Heading4.	14
	Referring to Controls, Views, and Menus.	16
	Radio Buttons and Checkboxes	16
	Drop-Down Lists and List Boxes	17
	Entering Information Into Fields.	17
	Buttons and Button Menus	18
	Right-Click Menu	18
	Views and Tabs	19
	Menus.	19
	Unnecessary Words and Their Alternatives	20
	Introducing Examples	21
	Writing Procedures.	22
	Launching Workbench Tools	22
	Bias-Free Writing.	24
	Typographic Conventions	25
	Use of Screen Shots.	27
	Cross-References.	28
	Cross-References to a Third-Party Help System	29
	Cross-References to Another Wise Book	29
	Help Navigation	29
	Standard Terminology	31
2	Indexing Standards	32
	Indexing Basics	33
	Summary of Indexing Standards	34
	Frame Indexing Issues	36
	Index for Help, Not Just the Book	37
	Indexing Guidelines	39
	Acronyms, Indexing	39
	Articles in Index Markers.	39
	Capitalization in Index Markers	40
	Cross-Book References, Indexing	40
	Cross-References in the Index	41
	Extensions, Indexing	41
	Grouping Index Items Under a General Term.	42
	Nouns, Indexing as Plural or Singular	42
	Page Ranges in Index Markers	42

Prepositions in Index Markers	42
Product Name, Indexing	43
Redirection in the Index	43
Special Characters/Formats in Index Markers	44
Syntax for Topic Headings	44
User Interface Elements, Indexing	44
Verbs in Index Markers	48
3 Wise Terminology	51
4 Tools and Templates	68
Tools for Creating Wise Documentation	69
FrameMaker Considerations	71
Setting Up FrameMaker	72
About Frame and Help Templates	74
Differences Between Frame Templates	76
Changing the Frame Templates	76
5x8 Format Considerations	77
Guidelines for Frame Template Conversions	78
Creating Conversion Documents.	78
Converting Books to a New Template	79
Checking the Document After a Template Conversion	81
A4 Page Size	83
Creating an A4 Template.	84
5 Documentation Structure and Formatting.	85
Formatting Text in FrameMaker	86
Formatting Headings	87
Proper Use of Headings.	87
Using H1s That Look Like H2s	87
Forcing a Line Break in Headings	87
Using Character Tags	89
Wide Tags	89
Using Change Bars.	91
Turning Change Bars On and Off by Book or Chapter	91
Clearing Change Bars in an Entire Document or Book	91
Removing and Applying Change Bars in Selected Text	92
Searching for Change Bars	92
Formatting Lists	94
Using and Formatting Notes	96
Anchoring Tables	98
Forcing Line Breaks	99
Creating Screen Shots and Placing Graphics	100
Setting Screen Colors for Screen Shots	100
Using PaintShop Pro for Screen Shots	101
Placing an Image in FrameMaker	101
Specifying the Format for Graphics in Help	103
Sizing Graphics for Help	105
Hyperlinks	106
Creating Links to URLs	106

Creating Links to External Help Topics	107
Using Conditional Text	110
How to Use the not for help Tag	110
Conditional Text Shortcuts	111
Inserting Temporary Author's Notes	112
Multilingual Production Requirements	113
6 Wise Documentation Sets	114
Wise Package Studio	115
Wise for Windows Installer	116
Wise for Visual Studio .NET	117
Wise Installation System	118
Wise InstallTailor	119
7 File Management	120
File Locations	121
Sharing Files	122
Archiving Documentation Files	124
8 Single Sourcing	125
Planning Conditional Documents	126
Managing Single-Sourced Books	127
Archiving Single-Sourced Books	127
Condition Tags in Single-Sourced Documents	128
Rules for Tagging Conditional Text	128
Creating a New Condition Tag	129
Creating Condition Templates	131
Product Name Variables	132
Text Insets	133
Text Inset Guidelines	133
Creating a Text Inset	134
Importing a Text Inset	134
Condition and Style Problems With Text Insets	135
Updating Text Insets	136
Archiving Documents That Contain Text Insets	136
Documenting the Visual Studio Integrated Editor	137
9 Producing Inline Help	139
Updating WfWI/WIE Inline Help	140
Linking to Online Help From Inline Help	141
Linking to SDK Help From WfWI Inline Help	142
10 Wise Documentation Deliverables	143
Deliverables	144
Deliverables from the Writer to Wise	144
Deliverables from Wise's Graphic Design	144
Preparing Documentation for Review	146
Documentation Review Guidelines	146
Delivering Beta Documentation	148

Producing the Beta Release Notes	148
Producing a Beta Manual	148
Creating Beta Help	149
Guidelines for ReadMes/Release Notes	151
Including Links to Previous Release Notes	151
Creating Release Notes for a New Version	152
Finishing Steps Checklist	153
Check and Update the Book's Properties	155
Reset the Document Numbering	155
Check and Update the Book's PDF Setup	156
Update the Copyright Date and Version Number	156
Search For and Fix Unresolved Cross-References	157
Hide Conditional Text	157
Changing the Book View	158
Check Pagination	159
Update the Book	160
Edit the Table of Contents	160
Format the Index	160
Converting to 5x8 Format	160
Converting to 7x9 Format	161
Converting to 8.5x11 Format	162
Housekeeping	164
Cleaning Up Graphics Directories	164
11 Creating .PDF and PostScript Files	165
Setting Up Acrobat Distiller	166
Creating a .PDF File	167
Creating a PostScript File for Commercial Printing	171
Deliverables to the Printer	171
Installing the Generic PostScript Printer Driver and Setting It Up	172
Saving the Book as PostScript	172
Testing the PostScript File	174
Printing a Hard Copy for Proofing	174
Sending the Book to the Printer	175
Reviewing the Printer's Proof	175
Creating a .PDF File for Quick Printing	177
Specifications for Quick Printing	178
Delivering Files to West Wind Litho	179
12 Creating Help With WebWorks	180
Preparing Frame and .H Files for WebWorks	181
Setting Up the "Copyright and trademarks" Popup	181
Setting Up HelpHeader and HelpFooter Markers	182
Preparing Graphics	182
Preparing Headings	182
Connecting Special Links	182
Mapping Context-Sensitive Help	183
Tips for Mapping Specific Kinds of IDs	185
Using WebWorks Publisher	188
Setting WebWorks Publisher Preferences	188
WebWorks Templates	188

Generating Help Using WebWorks Publisher	195
Specifications for WebWorks Publisher Help Projects	196
Generating an Alias File	199
Installing WebWorks AutoMap	200
Using WebWorks AutoMap	201
Locating WebWorks AutoMap Files	201
Setting WebWorks AutoMap Preferences	202
The WebWorks AutoMap Window	202
Creating or Editing AutoMap Jobs	203
Updating AutoMap Jobs	209
Scheduling AutoMap Jobs	211
Using AutoMap to Create PDFs	211
Troubleshooting AutoMap and Publisher	212
Importing Help Files into Visual SourceSafe	215
Adding Bread Crumbs to a Help File	217
13 Testing and Troubleshooting	219
Testing Help	220
Testing Fonts in Help	220
Testing Hyperlinks in Help	221
Testing Context-Sensitive Help	221
Testing Inline Help	222
Testing the Help Table of Contents	222
Testing the Help Index	223
Reporting Help Problems	223
Suggestions for Dividing the WfWI Help Testing	224
Final Testing of Documentation Deliverables	225
Troubleshooting	226
Verifying Tutorial Files in the Product	226
Troubleshooting Help	226
Troubleshooting the Frame Book	226
Missing Fonts Error in Frame	227
Troubleshooting the .PDF	228
Old RoboHelp Issues	229
Enabling All Options in Package Distribution	231
14 Department Procedures	232
Installing Product Builds	233
Tips for Installing and Uninstalling Wise for Windows Installer	233
Tips for Installing and Uninstalling Wise Package Studio	234
Reporting Time	235
Using Littlebuggy	236
Creating an Incident in Littlebuggy	237
Creating Littlebuggy Queries	239
Common Littlebuggy Queries for Tech Writers	239
Adding Columns to the Littlebuggy Query Table	240
FrameMaker Features You Need to Know	241
Index	242

Chapter 1

Standards for Writing Wise Documentation

As of August 2003, we adhere to the Abreve® methodology for writing and editing. We have incorporated Abreve standards into this guide.

Abreve is a proprietary methodology developed by ArchiText Inc. It increases usability by increasing clarity, conciseness, and consistency. For details, visit www.architext-usa.com.

Topics include:

- ◆ [Focus and Audience.](#)
- ◆ [Sentence Construction.](#)
- ◆ [Chapter Openings.](#)
- ◆ [Topic Introductions](#)
- ◆ [Referring to Controls, Views, and Menus.](#)
- ◆ [Unnecessary Words and Their Alternatives.](#)
- ◆ [Introducing Examples.](#)
- ◆ [Writing Procedures.](#)
- ◆ [Bias-Free Writing.](#)
- ◆ [Typographic Conventions.](#)
- ◆ [Use of Screen Shots.](#)
- ◆ [Cross-References.](#)

Focus and Audience

Our audience consists of:

- ◆ System administrators who package, customize and deploy applications for use within an organization.
- ◆ Professional developers who create applications for use in organizations or for sale to third parties.

They are advanced users, therefore we can assume that they understand common computer terms and know how to perform standard Windows tasks.

Writing for Our Audience

- ◆ Use common Windows terminology.
- ◆ Do not painstakingly describe how to perform standard Windows tasks. (Example: Docking and undocking windows, dragging and dropping items, and so on.) Just tell the user that the task can be done.
- ◆ Avoid explicit steps for simple operations. Example: Instead of describing how to use the Browse button, just say "In X, specify a .REG file."
- ◆ If a feature behaves just like Windows Explorer, don't write a procedure for it; just say that it acts like Windows Explorer.
- ◆ Avoid constructions such as "It is very important to follow the steps closely."
- ◆ Write every topic as though the person reading it is a new user of our product. That doesn't mean you have to repeat information that occurs in another place in the book. It just means that you might need to refer the user to that other information.
- ◆ Write every topic as though it is the only topic that the user will read. Don't assume they start at the beginning of the book and read forward, or even that they start at the beginning of the chapter. This is even more applicable when the book is converted to help topics. Avoid the following types of constructions: "As always, you should use variable substitution here," or "As we've mentioned before, you can always use variable substitution," "As we discussed earlier," "In previous chapters," and so on.

Sentence Construction

Active Voice, Second Person, Present Tense

- ◆ **Do use** active voice, second person, present tense. Example:

(You) Complete the WebDeploy page if you plan to distribute your application over the Internet.

- ◆ **Don't use** passive voice. Example:

The WebDeploy page should be completed in order to distribute your application over the Internet.

- ◆ **Don't use** first person. Example:

We'll now discuss completing the WebDeploy page to distribute your application over the Internet.

Exception: It is appropriate to use first person in recommendations. Example:

We do not recommend editing the predefined templates directly.

- ◆ **Don't use** future tense. Example:

Completing the WebDeploy page will let you distribute your application over the Internet.

When you click Compile, the installation will be saved and compiled.

Exception: When you're clearly talking about something that will happen in the future. Examples:

When the user starts Wise Package Studio in the future, they will be logged on automatically.

Customize configuration settings that will be used each time SetupCapture is run.

- ◆ **Don't use** past tense. Example:

After you completed the WebDeploy page, the script changed to include new script lines.

Imperative Forms

Use imperative forms in procedures. Examples:

- ◆ Click OK.
- ◆ Specify a directory for the files.
- ◆ Click Change to open the Repackage Advanced Settings dialog, where you specify SetupCapture preferences.

Long Sentences

Edit long sentences.

Before:

If you previously added conditions to the Features page in Installation Expert, and then you added items to the condition using the Current Feature drop-down list at the top of Installation Expert pages, then some of the components on the Components tab already have conditions set. (46 words)

After:

Some components on the Components tab already have conditions set if you previously added conditions to a feature on the Features page, then added items to the condition.
(28 words)

Chapter Openings

Make chapter introductions as short as possible. Use 1 or 2 very brief paragraphs summarizing the chapter, then add a list of cross-refs to H1s in the chapter. Occasionally, you might add a cross-ref to an H2 where appropriate. Sometimes, you can add extra text to the cross-refs that further clarify what the user will learn.

- ◆ If there is crucial information in the chapter opening, move it to its own section named "About X" or something similar.
- ◆ In general, don't index anything in the chapter opening. However, you can cross-reference to it because it contains a TOC to the rest of the chapter.

Format for Cross-Ref List

- ◆ Always introduce the cross-ref list with "Topics include:" — don't use any other variant.
- ◆ Use the AlsoSee tag for the "Topics Include" line.
- ◆ Use the B1 Bull1 tag for the list of cross-refs. Use the Section only cross-ref format.
- ◆ Apply the help navigation condition to the list of cross-refs. <<<We should discuss this. Val questioned the need for showing the x-refs in the book. JK>>>
Exception: In a single-sourced book, when the list of cross-refs contains one or more conditional items, do not apply the help navigation condition.

Topic Introductions

- ◆ Avoid restating the heading in the first topic sentence. This may be choppy in some cases. Try to maintain flow of paragraph also, however. Try to use an introductory sentence that adds crucial detail not found in the heading.

Example: If the heading is "Adding Registry Entries", the topic sentence could be "Use the Registry page to add registry keys and values to the destination computer." This adds 3 pieces of information: that "entries" means "keys and values", that you use the Registry page, and that the items are added to the destination computer.

- ◆ If the introductory material of the topic contains 2 or 3 paragraphs of conceptual material before the procedure starts, consider moving that conceptual material into its own topic. Ideally, conceptual material should be in one topic, with the procedure in another.

Bad:

(Heading) Checking the Value of an Environment Variable

(Text) This section describes how to create a condition that checks the value of an environment variable.

Topic Heading Guidelines

These standards are based on recommendations in *Single Sourcing: Building Modular Documentation* by Kurt Ament.

Use a consistent heading syntax that indicates the topic type and the kind of information the topic contains. Per Kurt Ament, "...The more consistent your heading syntax, the more predictable your headings become. The more predictable your headings, the easier it is for users to find exactly the information they need."

Topic Type	Syntax
Definition lists	◆ Types of Conflicts
Examples	◆ Sample Command Line Option ◆ Examples of WiseScripts You Run From an .MSI
Information and concepts, also chapter titles	Use nouns or the following forms: ◆ About the Share Point Directory ◆ Merge Modules and Transforms ◆ Translation ◆ How Self-Registration Information is Captured ◆ When to Isolate Files
Itemized Lists	◆ Installation Requirements
Procedures	Answer the question "How?" with a gerund. ◆ Adding an Environment Variable ◆ Installing Services
Processes	Similar to procedures but try not to start with a gerund. Or just use "Process". ◆ Problem Solving ◆ Conflict Resolution Process
Troubleshooting scenarios	Describe the problem, not the solution. ◆ Troubleshooting Compile Errors ◆ Troubleshooting: When Custom Actions Don't Work

The following is from *Single Sourcing: Building Modular Documentation*.

Headings should answer specific questions as directly as possible. When possible, use a heading syntax that mirrors the syntax of the question.

Question	Heading
Who should use AcmePro?	Who Should Use AcmePro
What does AcmePro do?	What AcmePro Does
When should I upgrade AcmePro?	When to Upgrade AcmePro
Where is AcmePro executable?	Location of Important Files
Why should I use AcmePro?	Reasons for Using AcmePro
How do I install AcmePro?	Installing AcmePro * * This differs from Kurt Ament, who uses the infinitive form. We try to reserve the infinitive form for subheadings that introduce procedures.

Single-word Headings

Avoid single-word headings whenever possible. Usually these are H2 or H3 headings, and they make sense in the book, where the reader can see the context of the preceding topic. However, when these are converted to individual help topics, the context is lost and the heading is confusing when listed in the Help TOC.

Bad:

Example

Better:

Command Line Option Example
Example: Using Command Line Options
Examples of Command Line Options

Also see [Formatting Headings](#) on page 87.

Use of Heading4

Introducing Procedures

Use the H4 Heading4 paragraph style to introduce procedures. It has an autonumber format that resets the autonumbering in the Numbered style to 1. When introducing a procedure, use the form *To do this*:

When an H4 Heading4 introduces a procedure, only initial cap the "To" and proper nouns or product UI elements. Use a colon at the end of the heading. Example: To start ConflictManager:

- ◆ Use an H4 to introduce a procedure when 3 or more short paragraphs, or fewer long paragraphs, precede the procedure. An alternative is to reduce or move the conceptual information out of the procedural topic. See [Topic Introductions](#) on page 13.
- ◆ Do not use an H4 to introduce a procedure when there are fewer than 2 or 3 paragraphs preceding it. Use your judgement as to how much introductory text is too much. In some cases, you can use a blank line to separate the procedure from the

preceding paragraph. Add a nonbreaking space (Ctrl+space) so that the blank line will not be ignored in the conversion to help.

- ◆ When a topic contains multiple procedures, use multiple H4s to introduce them. This is sometimes necessary if you can perform multiple tasks from one dialog, yet help can only call one topic for the dialog. Example:

(H2) Editing File Details

(H4) To edit attributes for a single file:

(H4) To edit attributes for multiple files:

Headings Within a Topic

Use H4 for subheadings to organize text within topics and facilitate scanning. Be aware that H4 paragraphs do not become separate help topics.

When an H4 Heading4 is used as a heading, initial cap as you would for other headings. Do not use a colon at the end.

Referring to Controls, Views, and Menus

Controls are checkboxes, radio buttons, fields, list boxes and buttons that appear on dialogs.

General Guidelines

- ◆ **Bold.** Use bold for control names. Consider eliminating bold for non-control names. If you need the bold, consider putting the non-field name on its own separate line.
- ◆ **Ellipsis.** Do not use to shorten a reference to a long control name. Spell out the entire control name. It's unfortunate, but the documentation must match the UI, however long. This is particularly important for localization.

Instead of:	Use:
If you marked the Don't update or recompress files ... option	If you marked the option Don't update or recompress files when saving (.MSI/.MSM only)

Radio Buttons and Checkboxes

Try to eliminate the "the" and "control-type" that surround the control name.

In a Paragraph or Numbered Step:

Instead of:	Use:
Mark the Add a digital signature option.	Mark Add a digital signature.
Mark this checkbox to leave the component installed.	Mark this to leave the component installed.

In Text Under a Bold Bullet:

Instead of:	Use:
◆ Leave installed on uninstall Mark this checkbox to leave the component uninstalled...	◆ Leave installed on uninstall Mark this to leave the component uninstalled...

Exceptions:

When referring to the control in text:

Mark the **Add a digital signature** option if you marked the **Y** option on the X page.

If the paragraph that follows the bold bullet is long, include "checkbox" or "option" in sentences at the end of the paragraph. Example:

- **Assign serial numbers after import**
Mark this to assign licenses to these users immediately. <More text in between.> If you clear this checkbox, the users are added to the Unassigned group without license assignments.

Drop-Down Lists and List Boxes

- ◆ On the Files page, select **Current Feature** > All Features.

- ◆ Select **Current Feature** > All Features. (drop-down list)

<<<I find this nomenclature for drop-down lists confusing. You really don't select Current Feature. You only select a value from the drop-down list. Would it not be better to treat drop-down lists like fields.

Before:

From the **Method** drop-down list, select **Rebuild using existing index file (not corrupt)**

After:

From **Method**, select **Rebuild using existing index file (not corrupt)** SS>>>

- ◆ Select **Merge Modules** > MDAC. (titled list box)

- ◆ From the left list box, select MDAC (untitled list box)

- ◆ There is still a problem if an option in a drop-down list does not have initial caps, and you have to put it in the middle of a sentence. Try to rewrite sentence so that the option name is at the end of the sentence.

Example:

To create an environment variable, in the Environment Variable Details dialog, select **Operation** > Set value on install and delete on uninstall.

If you MUST add more text after the option name, then bold the option text.

Example:

To create an environment variable, in the Environment Variable Details dialog, select **Operation** > **Set value on install and delete on uninstall** and then click OK.

- ◆ <<<< not sure about the above rules. Seth and JoAnn to review.>>>

Exception:

1. From **Active Project**, select a project.

If this installation contains multiple releases, the **Release** drop-down list appears.

Entering Information Into Fields

In **Name**, enter the installation name.

Exceptions:

- ◆ You can reassign a user to a different group by changing the **Group** field in the right pane or by dragging the user to the group name in the left pane.

- ◆ Example: The **File Name** field in Project Setup provides the value for the variable [FileName].

- ◆ You set the status by changing the **Status** field in Project Setup.

- ◆ When you mark this option, the **Location** field appears.

- ◆ After you execute a manual test case, enter the result in the **Status of Test Case** field of the test case.

- ◆ For an explanation of the **Status of Test Case** field, Status column, and status icons, see X.

- ◆ Double-click any test item and add comments in the **Test Details** field of the dialog that appears.

- ◆ The **Copy Installation File To** checkbox enables **Pathname**.

Instead of:	Use:
The Notes field provides space for you to enter additional information about the project.	In Notes , enter any additional information about the project.
The Process Notes field provides space for you to enter a brief description of this process.	In Process Notes , enter a brief description of this process.

Buttons and Button Menus

Omit "the" and "button" surrounding button names.

Example: Click Add Files.

Exceptions:

- ◆ If you need to explain the state of a button (example: "The Add button is disabled if ...")
- ◆ The following construction: When you use the Define button in Process Templates Setup to access this dialog...
- ◆ If multiple buttons on one page start with the same word ("Add", "Add Files", "Add Contents"), use "the" and "button" and also specify the location of the button.
- ◆ Click the Add button at the right of the Services page.

Note

Use "at the right" rather than "to the right." The latter sounds like it's next to the Services page instead of on it.

- ◆ The following construction: Mark this checkbox to add the ! switch to the command line, which removes the Cancel button from the installation dialogs.
- ◆ The Advertising Options button appears only when the **Install Mode** in the Define Command Line dialog is Advertised.
- ◆ If the Correct All button is enabled, click it to have all the listed issues corrected automatically.

Button with Drop-Down Menu

Example:

Click Add and select Import.

Exception:

Click Add to Project and select a project from the button menu. (Not button's menu.)

Right-Click Menu

- ◆ Right-click the process icon and select Duplicate.
- ◆ Right-click in the tools area and select New.
- ◆ Use the Expand and Collapse commands in the right-click menu to determine how the folders are displayed.

Instead of:	Use:
To copy data from a single cell of a report so you can paste it into a document, right-click in the cell that contains the data to copy. From the right-click menu, select Copy.	To copy data from a single cell of a report so you can paste it into a document, right-click in the cell that contains the data and select Copy.
Right-click the process icon and select Duplicate from the right-click menu.	Right-click the process icon and select Duplicate.

Views and Tabs

Views

- ◆ Go to Setup Editor.
- ◆ Go to Installation Expert.
- ◆ Go to MSI Script.
- ◆ Select Setup Editor > Dialogs tab.
- ◆ Select Installation Expert > Registry page.
- ◆ Select MSI Script > Execute Immediate tab.
- ◆ Select Setup Editor > Product tab > Properties icon.
- ◆ Go to the Registry page. (If they are already in Installation Expert.)
- ◆ Click the Dialogs tab. (If they are already in Setup Editor.)

Multi-tab Dialog

- ◆ Click the General tab.
Use this if the user is in a multi-tab dialog and you need to tell them to go to another tab.
- ◆ Double-click the custom action and complete the Details tab:

Menus

- ◆ Select Edit menu > Preferences.
- ◆ Select Edit menu > Preferences > General tab and mark **Create backup copy on save**.
- ◆ To make backups, select Edit menu > Preferences > General tab and select **Create backup copy on save**.

Unnecessary Words and Their Alternatives

Condense or eliminate the following:

- ◆ a number of -> several
 - ◆ all of the
 - ◆ if you want -> consider putting (Optional.) before the sentence.
 - ◆ in order to -> to
 - ◆ you want -> consider putting (Optional.) before the sentence.
 - ◆ you have the option to -> you can
 - ◆ you can choose to -> you can
 - ◆ automatically (sometimes this is needed, sometimes it is not)
 - ◆ designed specifically to
 - ◆ fill in, fill out -> enter, populate, or pre-fill, depending on context
 - ◆ for more information on
 - ◆ for details on
 - ◆ free up -> free
 - ◆ in the event that -> if, when
 - ◆ speed up -> speed
 - ◆ check to make sure that -> verify that
 - ◆ desired -> omit unless it's really necessary
 - ◆ select an option to determine -> specify
 - ◆ is designed to -> omit. Instead of "ConflictManager is designed to detect conflicts", say "ConflictManager detects conflicts."
 - ◆ When you are finished -> When you finish
 - ◆ After you have created -> After you create
 - ◆ large in size -> large
 - ◆ for you (as in "a folder will be created for you") -> omit
 - ◆ while the installation is running -> during installation
 - ◆ steps for selecting:
 - Select the group to add a user to -> Select a group
 - Select an item to edit and click Details -> Select an item and click Details
 - Select the directory that you want to add the file to and... -> Select a directory and...
- however:
- Select the file to copy from and click...
- ◆ filled in automatically -> pre-filled or populated
 - ◆ stored in, located in (as in "...is stored in the sample directory) -> in (unless the storage is important to the context)

Introducing Examples

Use only "Example:" or "Examples:" to introduce examples. **Do not use** "For instance", "For example," "Here is an example."

If the "Example" appears in the middle of a paragraph, put it in parentheses (example: like this). For clarity, you can put the example on a new line.

Example: like this.

If the example is really long, put Example: on one line, and the example itself on the next line.

Example:

Here is an example of a really extraordinarily amazingly long example.

Cultural Considerations in Examples

Previously, we have used such words as WidgetWare to use as example text for application or company names. Because those won't translate well, use the following example names:

- ◆ Application Names: Application1, Application2, Application3, etc.
- ◆ Packages: Package1, Package2, Package3, etc.
- ◆ Company Names: Company1, Company2, Company3, etc.
- ◆ Various: Sample1, Sample2, Sample3, etc.

Writing Procedures

See [Creating Screen Shots and Placing Graphics](#) on page 100 for an example that illustrates a typical procedure. It also shows what styles to use in a procedure.

If you're writing a procedure that's very simple, or that has multiple branches that a user could choose, use bullets instead of numbers. This form is not meant for procedures with several sequential steps. See the following examples.

To view a component error message:

In Setup Editor > Components tab, right-click the red component icon and select Show Errors.

To select script lines:

- ◆ To select one script line, click the line.
- ◆ To select multiple consecutive lines, click on a line and drag up or down. You can also click on the first line then Shift+click the last line.
- ◆ To select multiple non-consecutive lines, hold down the Ctrl key while you click on lines.

Launching Workbench Tools

If the following samples do not fit the situation, refer to WBTools.fm or other tool-related chapters in the Wise Package Studio book for similar situations.

Basic

(The sentence in <> below would be optional.)

1. Do one of the following:
 - On the Projects tab, click the Run link to the right of the task or tool associated with ApplicationWatch. <The package that is created will be saved with the default project name.> This tool might skip dialogs or populate fields based on command line options defined in Process Templates Setup.
 - On the Tools tab, double-click ApplicationWatch.
2. If the Specify Target Installation File dialog appears, specify X and click Next.

Tools Like ConflictManager

1. In Wise Package Studio, do one of the following:
 - On the Projects tab, click the Run link to the right of the task or tool associated with ConflictManager.
 - On the Tools tab, double-click ConflictManager.
 ConflictManager opens.

More Complex

1. Do one of the following:
 - On the Projects tab, click the Run link to the right of the task or tool associated with Package Distribution. The .MSI associated with the current project will be distributed. This tool might skip dialogs or populate fields based on command line options defined in Process Templates Setup.

- On the Tools tab, double-click Package Distribution. On the Specify File to Distribute dialog that appears, specify an .MSI and click Next.
- In Software Manager, in the Package List, click a package. Then select Packages menu > Distribute.

Bias-Free Writing

Avoid writing that may show bias of any kind, but especially of nationality or gender.

- ◆ In examples, avoid using names. Instead, use User_1, User_2, and so on. If you must use names, use a variety of both male and female names.
- ◆ Avoid pronouns where possible.

Instead of:

The user can change his default settings.

Use:

The user can change the default settings.

- ◆ Avoid these awkward constructions: his/her, his or her, he/she, he or she. If you absolutely must use these pronouns, use "or" instead of the slash.
- ◆ Avoid the generic masculine pronoun. It is acceptable to use a plural pronoun even when the antecedent is singular.

Instead of:

This dialog lets an end user enter his name, company, and product ID.

Use:

This dialog lets end users enter their name, company, and product ID.

A user can specify how to validate his logon by selecting from a list of Windows NT domains.

A user can specify how to validate their logon by selecting from a list of Windows NT domains.

Typographic Conventions

The Abreve methodology advocates minimal use of special formatting. As a result, we have eliminated most of the character formatting we used to use. We continue to use bold for control names, primarily because the capitalization in the UI is inconsistent, and some control names are very long, making it hard to pick out the control name in text.

- ◆ Initial caps in text always follow the UI.
- ◆ To refer to a shortcut, format it like this: Press Ctrl+P.
- ◆ Minimize bolding of non-control names, or put it on a separate line.
- ◆ Use numerals in all cases unless the number is the first word in the sentence. If the number is the first word, consider reworking the sentence so it is not the first word.

Item	Convention/Appearance	Frame Character Tag
article and book titles	<i>Initial Caps, Italics</i>	emphasis
checkbox name	Bold	field
condition name	<i>Italic</i>	emphasis
document name	<i>Italic</i>	emphasis
drop-down list name	Bold	field
drop-down list option	Bold	field
emphasis - use rarely	<i>Italic</i>	emphasis
field contents	Bold	field
field name	Bold	field
list box name	Bold Do not bold the word <i>list</i> .	field
logical operators	AND, OR, XOR	
option	Bold	field
radio button	Bold	field
web address	blue (if you will make it a hyperlink)	Weblink

Formatting Controls in a Dialog

Before Abreve:

- ◆ **Application Name.** Enter the name of the application you are installing. This name appears in the desktop computer’s mobile device software (the Add/Remove Programs dialog, accessible from the Microsoft ActiveSync window).
- ◆ **Description.** The description you enter here appears to the end user in the Add/Remove Programs dialog before installation.
- ◆ **Name of Setup Files.** When you compile, several files are created, including an .INI, and .INF, and one or more .CABs. This .INF file is used to create the .CAB files needed to install on the mobile device. The .INI file is used by the .CABs during installation to the mobile device. The name you enter here determines the naming

convention of all the .CAB files and other related files. All .CAB files are named according to the following structure: [*Name of Setup Files.Processor Name.CAB*].

After Abreve:

◆ **Application Name**

Enter the name of the application you are installing. This name appears in the desktop computer's mobile device software (the Add/Remove Programs dialog, accessible from the Microsoft ActiveSync window).

◆ **Description**

The description you enter here appears to the end user in the Add/Remove Programs dialog before installation.

◆ **Name of Setup Files**

When you compile, several files are created, including an .INI, and .INF, and one or more .CABs. This .INF file is used to create the .CAB files needed to install on the mobile device. The .INI file is used by the .CABs during installation to the mobile device. The name you enter here determines the naming convention of all the .CAB files and other related files. All .CAB files are named according to the following structure: [*Name of Setup Files.Processor Name.CAB*].

Use of Screen Shots

In July 2002, we decided to eliminate unnecessary screen shots from our documentation. Although the decision was prompted by the need to reduce page counts, our research convinced us that cutting screen shots made sense from a usability standpoint. The consensus of several studies and discussion threads is:

- ◆ Excessive use of screen shots can detract from usability.
- ◆ Screen shots are less important for experienced users (our audience) than for novices.
- ◆ Screen shots are useful when there is a risk of confusion.
- ◆ Screen shots presented at critical moments in a procedure can help mark the user's progress.

Use the guidelines below to help you decide when to include a screen shot; include the screen shot only if you can answer yes. They are not absolute rules; use your judgement in each case.

- ◆ Would a screen shot here help the intended readers?
- ◆ Could this screen shot provide any information that isn't already written in the documentation?
- ◆ Can I add something to the screen shot to make it somehow more useful than the actual screen itself?
- ◆ Would the insertion of screen shots at critical stages of this procedure help users verify their progress?

When to Use Screen Shots

- ◆ To show a special icon, tool, or button. Unless you need to show the context in which the icon appears, show just the icon, and place it in the text instead of on its own line.
- ◆ To show an overview of the product interface, usually with callouts.
- ◆ To locate window elements and objects, but only when the UI is complex, confusing, or non-standard.
- ◆ To verify the screen state in tutorials and examples. Example: *When you complete the Details dialog, it should look like this: <screen shot>*
- ◆ To help mark a user's progress through a complex procedure.

See [Creating Screen Shots and Placing Graphics](#) on page 100.

Do Not:

- ◆ Insert every dialog a user encounters during a procedure.
- ◆ Insert an entire screen where only a piece (or a picture) will do.
- ◆ Use a screen shot just to show how a field entry should be formatted. Include the information in the text instead. Example: Instead of inserting an entire dialog just to show how a URL should be entered, say *Use the format: http://www.mysite.com/filename.msi.*

Cross-References

Cross-references should all have the same look and feel. Because in the .PDF and help, the cross-refs are hypertext, users should be able to recognize them instantly. Keeping them all in the same general format, rather than using a variety of cross-ref formats, lets user easily recognize these easy paths to further information. It also reduces localization costs.

For information on cross-reference formats, see <<<not sure where this will be>>>.

Cross-Reference Constructions

- ◆ See
Especially use "See" to eliminate redundancy in this type of sentence: "For information on detecting and resolving conflicts, see *Detecting and Resolving Conflicts*." Instead, just say "See *Detecting and Resolving Conflicts*."
- ◆ For details, see
Use if you think the sentence needs this extra information.
- ◆ For information on
Use to rewrite the following types of constructions:
 - See the description of the Level field in Configuring a Feature Using the Feature Details Dialog on page x.
 - To learn how to check the version of the IIS server, see Setting a Requirement on the System Requirements Page.
 - A properties dialog appears for each type of control, which are described in Basic Control Settings on page x.
 - For instructions on handling dialogs on the destination computer, see Working With Dialogs on page x.
- ◆ <<<Add one for the new "Step" x-ref after template redesign. JK>>>

Do Not Use These Constructions

- ◆ For more information...
- ◆ For hints...
- ◆ on the next page, in the next section, on the previous page, later in this chapter, and so on.
- ◆ See page 121.
- ◆ This is discussed in more detail on page 232.

References to URLs

- ◆ Search for "Interoperating with Unmanaged Code" in the MSDN Library (msdn.microsoft.com/library/).
- ◆ For additional language codes, search for "Table of Language Identifiers" and "Language Code and Scripts" in the MSDN Library (msdn.microsoft.com/library/).
- ◆ For information about digital signatures, search for "authenticode" on the VeriSign Web site (www.verisign.com).
- ◆ For information on Visual Basic, visit msdn.microsoft.com/vbasic/.

See [Creating Links to URLs](#) on page 106.

Cross-References to a Third-Party Help System

Several Wise products contain the Windows Installer SDK Help, to which the product documentation refers frequently.

- ◆ Preferred format:
See *Windows Installer and Logo Requirements* in the Windows Installer SDK Help.
- ◆ If additional explanation is needed, or if the topic title is not descriptive of what you want the user to find:
 - If the issue was found by a Microsoft validation module, see *ICE Reference* in the Windows Installer SDK Help for details on the issue.
 - Before you attempt to add components here, see *Windows Installer Components* in the Windows Installer SDK Help.
 - For a list of these special keys, see *User Policies* and *Machine Policies* in the Windows Installer SDK Help.

As of April 2003, references to the SDK Help are tagged as hyperlinks. The hyperlinks do not work in the .PDFs but they do work in the online help. For instructions on tagging the SDK topics as hyperlinks, see [Creating Links to External Help Topics](#) on page 107.

Cross-References to Another Wise Book

When a product's documentation consists of multiple books, you can create cross-references between books. In order for these links to work in the help, all referenced help files must be installed in the same directory as the main help.

These links do not work in the .PDF. If an end user hovers the mouse over one of these links, the finger icon will appear, but the link will not work. Because the links are not colored blue in the .PDF, we have to hope the user won't think to try the links.

Format:

See *Setting Up Tools* in the Wise Package Studio Help.

Where the italicized text is a hyperlink.

You create a cross-book reference in the same way as any other cross-reference, except you use the cross-reference format X-book.

Help Navigation

Because help is not linear, some of the continuity of the book is lost in the conversion to help. Therefore, we add navigation hyperlinks to topics to help the user find related topics. To do this, you decide where to add navigation, then insert a cross-reference to link to the desired topic. The navigation usually takes the form of an "Also see:" list at the end of the topic.

Guidelines

For help in deciding where to add help navigation, refer to a chapter TOC.

- ◆ The first topic of each chapter should contain hyperlinks to the H1 topics in that chapter. See [Chapter Openings](#) on page 12.
- ◆ Add navigation to any topic that introduces a series of topics, or that serves as an overview.
- ◆ Add navigation between topics that are closely related. Example: The topics "Creating a Patch" and "What You Need to Create a Patch" should contain hyperlinks to each other. (We frequently add that kind of reference in the book, but it's not always appropriate or necessary.)

- ◆ Add navigation from low-level topics back to the main or introductory topic.

Format for Help Navigation Links

When you add the navigation links in the Frame files, use the cross-reference format **Section only**, and apply the conditional text tag **help navigation**. Show this text while you are writing and editing the book. Hide this text for reviews and for producing the book .PDF and printed manual. For details, see [Using Conditional Text](#) on page 110.

For the "See" and "Also see" text, use the AlsoSee paragraph tag and for the links that follow, use the BT BodyTight style tag. Example:

Also see:

[Help Navigation](#)
[Cross-References](#)

When there is only one hyperlink, don't put it on a separate line; make it a complete sentence. Example:

Also see [Help Navigation](#).

Standard Terminology

<<< What to do with these? They don't seem to belong in the Terminology section.

JK>>>

- ◆ your installation - in most cases, change to "the" or "an."
- ◆ your application. Not your product, or your software.
- ◆ in the directory
- ◆ on the tab
- ◆ on the Files page
- ◆ on the Properties dialog (This is contrary to the way we've written this in the past, so these cases must be fixed.)
- ◆ Complete the dialog: (Introduces a series of controls in a procedure.)
Use instead of "Fill out the dialog as follows," "Configure the whatever dialog," "Fill out the fields on the whatever dialog" and other variations.

If there are not many fields, you could say "Complete the dialog and click OK" in one step. But if there is a long list of fields, it's better to put the Click OK after them as a separate step.

Chapter 2

Indexing Standards

This section contains standards for indexing Wise Solutions documents. Following these standards should make the various indexing styles of each writer more consistent and should reduce the amount of time it takes to review and edit the index.

Some material in this chapter was suggested by concepts included in *Indexing: A Nuts-and-Bolts Guide for Technical Writers* by Kurt Ament. No material was reproduced exactly.

Topics include:

- ◆ [Indexing Basics.](#)
- ◆ [Summary of Indexing Standards.](#)
- ◆ [Frame Indexing Issues.](#)
- ◆ [Index for Help, Not Just the Book.](#)
- ◆ [Indexing Guidelines.](#)

Indexing Basics

If you read nothing else in this section, read this topic and the [Summary of Indexing Standards](#) on page 34.

In general, do not index interface elements only. For each topic in the book that you are indexing, take a deep breath, really look at the topic, skim it, and determine:

- ◆ What goal would this section help a user accomplish?
- ◆ In thinking of that goal, what words would come to the mind of the user?

This might not be the easiest way to index, but it will prove more fruitful for the user who is using your index.

Also, try to index multiple subjects if you are unsure what the user would look up first.

Example: If the user's goal is to "convert a VB project to a Windows Installer installation", what would the user look up first?

Here are some possibilities:

- converting, VB project
- importing, VB project
- VB project
- Visual Basic project

The user should find helpful markers by looking up any one of the above words.

Summary of Indexing Standards

Following is a summary of the indexing guidelines described elsewhere in this section. If you read nothing else in this section, read this topic and [Indexing Basics](#) on page 33.

Acronyms

For common acronyms (i.e. HTML), index the acronym only. For less common acronyms (example: MSDE), index the acronym as the main marker, and use a "See" reference to index the spelled out version of the acronym. See [Acronyms, Indexing](#) on page 39.

Articles

Avoid articles, such as "a", "an" and "the". See [Articles in Index Markers](#) on page 39.

Capitalization

User lowercase. Exceptions are proper nouns or following the UI of the product. See [Capitalization in Index Markers](#) on page 40.

Cross-Book References

In cases where it is likely that the user might look in the wrong book for a particular term or feature, include a cross-book reference, which points the user to a different book or help system. See [Cross-Book References, Indexing](#) on page 40.

Cross-References

Optionally, use cross-references ("See also" references) when another related topic may help the user. These are usually reciprocal. These are optional because the topic itself usually contains these kind of cross-references. See [Cross-References in the Index](#) on page 41.

Extensions

When indexing extensions, index them without the period so that they appear alphabetically in the index. Use a "See" reference to point users from the beginning of the index to the actual index marker; for instance ".EXE:See EXE". See [Extensions, Indexing](#) on page 41.

General Terms

Avoid indexing lists of items under a general term. Example: indexing all page names under the marker "Installation Expert". See [Grouping Index Items Under a General Term](#) on page 42.

Nouns, Plural or Singular

Use singular in general, reserving plural only for those items that are discussed in a mostly plural sense, such as "preferences". See [Nouns, Indexing as Plural or Singular](#) on page 42.

Page Ranges

Do not use page ranges (i.e. files:adding 13-21). See [Page Ranges in Index Markers](#) on page 42.

Prepositions

Avoid prepositions unless they are necessary for meaning. See [Prepositions in Index Markers](#) on page 42.

Product Name

In general, do not index the name of the product you are working on. The users know what product they are using. See [Product Name, Indexing](#) on page 43.

Redirections

Use "See" references to refer to the main marker that covers the current marker.

Example: Make "custom action" the main marker with all subentries, and index "action" with a "See custom action" reference. See [Redirection in the Index](#) on page 43.

Referencing General Terms

Avoid indexing lists of items under a general term. Example: indexing all page names under the marker "Installation Expert". See [Grouping Index Items Under a General Term](#) on page 42.

Special Characters/Formats

For WinHelp project, special characters cause problems in converting to help, but John's utility and our Word macros takes care of these. For HTML Help projects, the special character problem does not exist. However, do not use FrameMaker formatting that makes markers float to the top of a list of subentries ([script:aaa]), which causes conversion problems. See [Special Characters/Formats in Index Markers](#) on page 44 and [Frame Indexing Issues](#) on page 36.

Syntax for Topic Headings

When indexing, look to the headings for clues on how to index. Often you can rearrange the heading to get a good index marker. See [Syntax for Topic Headings](#) on page 44.

User Interface Elements

There are no hard and fast rules for indexing interface elements. Sometimes it is appropriate to index them as-is, and sometimes you should take a more goal-oriented approach. See [User Interface Elements, Indexing](#) on page 44.

Verbs in Index Markers

Main entries: Try to use the noun form of a verb; if that is not possible, use the gerund ("ing") form.

Subentries: Use the gerund form of the verb.

For guidelines on how and when to use verbs in index markers, see [Verbs in Index Markers](#) on page 48.

Frame Indexing Issues

- ◆ When entering an index marker, make sure the **Marker Type** is always set to **Index**. The **Marker Type** field can get reset automatically. You can check an individual file for problems of this nature by generating a list of all cross-references in the file.
- ◆ Place index markers at the beginning of a sentence only. This aids in localization.
- ◆ Do not place an index marker at the beginning of a heading line. Headings often have cross-reference markers, which makes it hard to find the index markers.
- ◆ Do not place index markers anywhere in a heading, as they might cause problems in help.
- ◆ When entering cross-reference ("see") markers, make sure that <\$nopage> appears at the beginning of the marker text for the cross-reference entry. In a marker with multiple entries, put the "nopage" entries last, because ALL entries following the <\$nopage> code will not have page numbers.
- ◆ Do not use Frame formatting that floats a subentry to the top of the list of subentries, such as "<\$nopage>Components: Also see Features [Features:aaa]". This causes a problem in help. <<<This was a problem in RoboHelp. Apparently it's not a problem in WebWorks. Need to decide if we want to use it. JK>>>

Index for Help, Not Just the Book

When you index, force yourself to think of topics the way they are presented in online help instead of the way they are presented in the book. In the book, sloppy or inadequate indexing is more forgivable. If you get the user to the first page of a section or chapter that discusses their topic, the user can continue paging through the book to find additional related topics. This is not as easy to do in online help. Having specific and descriptive index markers directed to the right place in every topic makes it easier for users to quickly find what they need in online help.

Do Create Specific Markers for Each Topic That Discusses an Entry

Example: Instead of just pointing users to the first page of the Files page description, make sure specific markers exist for each subsequent topic that discusses files. This sometimes leads to the unfortunate result of having multiple consecutive markers in the index leading to the same page in the book. Do it anyway, because in the online help, related topics are not on the same page and page numbers disappear.

Avoid:

file
 adding 23
 editing 25

Use this instead:

file
 adding 23, 24
 adding with wildcard 23
 configuring attributes 25
 deleting 27
 editing 25
 installing 23

Do Not Create Multiple, Undirected Entries

Multiple page numbers for an entry are not as useful as specific subentries. Sometimes multiple references are unavoidable but you should try to break them into subentries whenever possible.

Avoid:

variable parameters 482,
 483, 485, 487, 527

Use this instead:

variable parameters, using to call .DLL
 overview 482
 for DLL stored in binary table 483
 for installed file DLL 485
 for DLL file on destination machine 487
 tutorial 527

Do Place Index Markers Where They Belong

In some of our older books, a single index marker in the first paragraph of a topic contains entries that apply to text appearing much later in the topic. Avoid this because it causes the following problems:

- ◆ In HTML help, index hyperlinks take the user directly to the location of the index marker. Therefore, the user might follow a link to *runtime properties* and end up in a paragraph that discusses properties in general. Scrolling will eventually take the user to the correct location in the topic, but it's better to direct the user to right place.

- ◆ When revising the book, it's easy to overlook index entries that are not in the paragraph you're editing. Example: Suppose all the index entries for a topic are in the first paragraph. During revision, you delete the third paragraph. It is doubtful that you'll think to review the index markers in the first paragraph, so you will end up with obsolete references to the third paragraph that no longer exists.

Instead of putting all index markers for a topic in one place, insert each index marker at the beginning of the sentence containing the text it refers to. This will result in more index markers, but it is more useful to the user and helps eliminate obsolete entries.

Indexing Guidelines

This section contains an alphabetical list of guidelines on specific indexing topics. For a brief summary of each item, see [Summary of Indexing Standards](#) on page 34.

See:

[Acronyms, Indexing](#)
[Articles in Index Markers](#)
[Capitalization in Index Markers](#)
[Cross-Book References, Indexing](#)
[Cross-References in the Index](#)
[Extensions, Indexing](#)
[Grouping Index Items Under a General Term](#)
[Nouns, Indexing as Plural or Singular](#)
[Page Ranges in Index Markers](#)
[Prepositions in Index Markers](#)
[Product Name, Indexing](#)
[Redirection in the Index](#)
[Special Characters/Formats in Index Markers](#)
[Syntax for Topic Headings](#)
[User Interface Elements, Indexing](#)
[Verbs in Index Markers](#)

Acronyms, Indexing

For common acronyms (examples: HTML, PDF, SQL, and URL), index the acronym. For more obscure acronyms (examples: SMS and MSDE), index the acronym as the main marker, and include another marker that spells out the acronym and refers to the main marker. If the acronym is an extension, omit the period in the index marker; see [Extensions, Indexing](#).

Avoid:

HTML 5, 7
Hypertext Markup Language 5, 7

SMS 6, 8
Systems Management Server 6, 8

Uniform Resource Locator 14
URL 14

Use this instead:

HTML 5, 7

SMS 6, 8
Systems Management Server
See SMS

URL 14

Articles in Index Markers

Avoid articles as much as possible in index markers. There might be instances where you need an article, but I can't think of any.

Avoid:	Use this instead:
custom action	custom action
setting a property	setting property
setting a directory	setting directory

Capitalization in Index Markers

Follow these guidelines:

- ◆ Use lowercase for all words that are not proper nouns. For proper nouns, follow established conventions.
- ◆ If you are indexing a UI element of the product, follow the UI of the product.
- ◆ If a word is both a proper noun and in the UI of the product, the capitalization of the item might differ between the two. Ideally, you should try to write a bug making the product's UI match real life; however, if you are indexing, it's probably too late for that. In this case, follow the UI of the product.

In the example below, the UI of the product has a page named "Autoexec.bat". But when you look on your hard drive, you see a file named AUTOEXEC.BAT. The "Use this instead:" example below follows the UI of the product.

Avoid:	Use this instead:
AUTOEXEC.BAT page	Autoexec.bat page
File	file
Searching For	searching for
system search page	System Search page

Cross-Book References, Indexing

Cross-book references tell the user to look in a different book or help system for a given term. Use them if you think a user might inadvertently search the wrong help system for a term. Doing cross-book references requires much more thinking and more time than doing an ordinary index, but it would be helpful for our multi-product offerings that include several full books.

Example: Suppose a user is in the Windows Installer Editor help system, with the goal of converting a legacy Wise installation to Windows Installer format. It would seem logical for this user to look in the Windows Installer Editor book. They might look up "convert", "Windows Installer:convert to", "installation:convert to Windows Installer".

The problem is that these items are not indexed in the Windows Installer Editor book, because the "Import SMS Installer or WiseScript Installation" wizard is in the Wise Package Studio book/help. Ideally, to rectify this problem, you would include cross-book references when you expect that a user might look in the wrong book for information.

Avoid:	Use this instead:
<nothing>	converting to Windows Installer; refer to <i>Wise Package Studio help</i>
	installation converting to Windows Installer; refer to <i>Wise Package Studio help</i>
	Windows Installer converting to; refer to <i>Wise Package Studio help</i>

Cross-References in the Index

You can use “Also see” references to point users in the direction of a related or parallel marker. (Example: If a user looks up Components tab, the Features tab markers might help them also, since the two are similar.) “Also see” references should be reciprocal, which means that each one points to the other. On the other hand, “See” references are not reciprocal; a “See” reference really means “See instead”.

Doing these types of cross-references is an optional “nice-to-have”. Because these types of cross-references should already be included in each topic, any that you put in the index should be supplementary to what’s already there.

Acceptable	Better
Components tab 15	Components tab 15 Also see Features tab
Features tab 26	Features tab 26 Also see Components tab
MSI file	MSI file Also see WSI file
WSI file	WSI file Also see MSI file

Extensions, Indexing

When you index extensions, index just the 3-letter extension*, not the period that begins the extension. Formerly, we indexed all extensions starting with a period, which meant they all appeared at the beginning of the index. Because it’s likely that users will look up the extension by the first letter of the extension, we now index them in alphabetical order, which means omitting the period.

Under the Extensions section in the index, create one index marker that points the user to the alphabetized extensions. It should be something like “Extensions: see first letter of extension”.

Avoid:	Use this instead:
.EXE calling creating	Extensions extensions See first letter
	EXE calling creating

* Exception: .NET

Grouping Index Items Under a General Term

Do not index lists of specific items underneath a general term. This is best described with some specific examples. Example: Under the marker "custom action", do not index all the individual custom action names. Under the marker "Installation Expert", do not index all the page names.

Nouns, Indexing as Plural or Singular

In general, for nouns, use singular, unless for some reason the item you are referring to almost always is referred to plurally, such as "system requirements" or "preferences". Note that this is not a issue of which way is more correct; it simply provides a standard guideline so that writers can be consistent with already existing markers.

Avoid:	Use this instead:
files adding configuring attributes file editing installing	file adding configuring attributes editing installing

Page Ranges in Index Markers

Don't use page ranges in index markers. A page range gives a range, such as 12-16, instead of a single page. Frame allows you to do this, but it would not work well in help.

Prepositions in Index Markers

Avoid using prepositions in index markers. To keep index markers as short as possible, only include key words, unless the preposition is required in order to make sense.

Example: Suppose you are indexing the System Search page and want to create an index marker that tells users they can search for files on the destination computer. The marker, "files:searching for" has a totally different meaning than "files:searching". If you omit the preposition in this case, you lose the precise meaning of the index marker.

Avoid:	Use this instead:
installation adding files to configuring attributes of	installation adding files configuring attributes
network distributing	network distributing to
WiseUpdate how use	WiseUpdate how to use

Product Name, Indexing

In general, do not index the name of the product you are working on. The users know what product they are using. If they're looking for a topic such as "functions", they are not likely to look first under "Wise for Windows Installer".

Avoid:	Use this instead:
Wise for Windows Installer command line options features launching Professional Edition Standard Edition	command line options editions Professional Standard features, Wise for Windows Installer launching Wise for Windows Installer

Redirection in the Index

A redirection marker is one that starts with "See" and tells the user to look at an alternate term for the term being looked up. Example: Suppose the product has something called "custom actions". Will the user look up "custom actions" or will the user look up "actions"?

Instead of indexing both, pick one of the two terms to serve as the main marker and put all the subentries under it. Choose whichever term you think is more likely to be looked up. Add the other term to the index, but put a "See" reference on the next line, referring the user to the main marker.

Avoid:	Use this instead:
action	action
about	See custom action
adding	
editing	custom action
types	about
	adding
custom action	editing
about	types
adding	
editing	
types	

Special Characters/Formats in Index Markers

Special characters in index markers used to cause problems when converted to WinHelp, however, they do not cause problems in HTML help.

One formatting problem causes problems converting to HTML help, but did not cause a problem for WinHelp. FrameMaker index markers that contain the code [script:aaa] cause errors when the Frame file is converted to HTML help, so avoid using this code. If you find one, remove it. See [Frame Indexing Issues](#).

Syntax for Topic Headings

If topic headings are well-structured, it is easy to index them. See the following examples.

Headings:	Index Markers:
Windows Installer 16	Windows Installer
Introduction to Windows Installer 17	about 16
How Windows Installer Works 18	advantages 19
Advantages of Windows Installer 19	introduction 17
	how it works 18
Preparing for Software Updates	Updates, preparing for
	Upgrades, preparing for
	Software updates
	preparing for

User Interface Elements, Indexing

This section provides guidelines for indexing each type of interface element you might encounter in Wise products.

See:

[Tool/Wizard Names, Indexing](#)
[Feature Names, Indexing](#)
[Page Names, Indexing](#)

[Names of Views, Indexing](#)
[Menu Commands, Indexing](#)
[Dialog Names, Indexing](#)
[Control Names, Indexing](#)

Tool/Wizard Names, Indexing

Index wizards and tools with their formal marketing name, as well as with key words that the user would look up based on how the tool meets the user's goal.

Avoid:

Validation Wizard 32

Use this instead:

Validation Wizard 32
 installation
 finding errors in 32
 Windows certification 32

Feature Names, Indexing

Index formal feature names (example: MSI Script). For less formal names, or for features not named at all, like Change Source Directories, don't index by name. Index by noun. In rare cases, also index by dialog name; see [Dialog Names, Indexing](#) on page 47.

Avoid:

Change Source Directories 32

Use this instead:

Change Source Directories dialog
 source directories
 changing 32
 directories
 changing source 32
 files
 changing source directory 32
 relative pathnames 32
 UNC-based pathnames 32

Page Names, Indexing

Index all Installation Expert page names. Also index nouns that users might associate with the page. Example: On the System Search page, you can search for components, directories, files, .INI content, or registry entries. Therefore, keep in mind that the user's goal might be to search for a directory. In that case, they might look up "directory:searching" but probably won't look up "System".

Do not index page names under the view they appear in. Example: Don't have a marker for "Installation Expert" which has subentries of every page name.

In the example below, notice that "searching" is indexed, because it is a verb with a very specific meaning; see [Verbs in Index Markers](#) on page 48.

Avoid:	Use this instead:
Installation Expert	component
System Search page	searching for
page	directories
System Search page	searching for
System Search page	files
	searching for
	INI file
	searching
	registry
	searching
	searching destination computer
	System Search page

Names of Views, Indexing

Index view names briefly, pointing users only to topics that discuss the view in general. Users generally look up words relating to their immediate goal. Unless their goal is specific to the view, such as using the view, they will probably not look up the view name. Therefore, do not index each subview or each function below a view.

Avoid:	Use this instead:
Installation Expert	Installation Expert
Add/Remove Programs page	about
Dependencies page	customizing
Dialogs page	using
Digital Signature page	navigating
Exclusion page	resetting pages
Etc.-->	

Menu Commands, Indexing

In general, do not index standard menu commands such as Copy and Paste, but make sure that non-standard, product-specific menu commands have some kind of index marker, even if the wording differs slightly. Ideally, if the user sees a non-standard menu command, and wants more information on it, the user should be able to look up the menu command in the index, and should at least be able to find something similar to the menu command.

If the menu command starts with a verb, try to use the gerund form of the verb for the index marker. Example: Instead of "Check for Updates", use "checking for updates". At the very least, the index entry should start with the same word or a form of the same word as the menu command.

If the menu command does not start with a verb, and it spawns a dialog of the same name, index the dialog. Example: ConflictManager contains a menu command called Network Index Properties. In that case, you cannot use a verb, so index the dialog it spawns, which is "Network Index Properties dialog." See [Dialog Names, Indexing](#) on page 47.

Avoid:	Use this instead:
---------------	--------------------------

Convert Source Paths 43 Convert Source Paths dialog 43 converting source paths 43 to UNC 44 to relative 45	converting source paths 43 to UNC paths 44 to relative paths 45
---	--

Network Index Properties 11

Network Index Properties dialog 11

Dialog Names, Indexing

In general, do not index dialog names. Users tend to look up words (mostly nouns, and sometimes verbs) that have to do with their immediate goal. There might be exceptions to this rule if the dialog has a unique name that users are likely to look up. If a menu command (i.e. Network Index Properties) has the same name as the dialog (i.e. Network Index Properties dialog), simply index the menu command, not the dialog name, for brevity's sake.

This does not apply to wizard dialogs, those dialogs that appear on the destination computer during installation. It is OK to index those.

When you are editing the index, watch for instances where markers with virtually identical wording are placed directly next to each other, as in the example below. In these cases, one of the markers can safely be eliminated.

Avoid:	Use this instead:
---------------	--------------------------

Call DLL From Installation 56 Call DLL From Installation dialog 56	Call DLL From Installation 56
---	-------------------------------

Convert Source Paths dialog 43 converting source paths 43	converting source paths 43
---	-------------------------------

Install MSI From Destination 89 Install MSI From Destination dialog 89	Install MSI From Destination 89
---	---------------------------------

Network Index Properties 66 Network Index Properties dialog 66	Network Index Properties 66
---	-----------------------------

Control Names, Indexing

Controls include drop-down lists, fields, list boxes, checkboxes, buttons, radio buttons, group boxes, etc. Sometimes it may be tempting to simply index every control in a dialog instead of thinking about user goals. Do not give in to this temptation. In most cases, think about what goal the checkbox/radio button/drop-down list would solve for the user, and start your indexing process from there.

If the control name is very specific and unique, such as “Credentials File”, “Listbox Compatibility Mode”, “Distribution Method”, “Previous MSI Path”, or “Change All Paths to Relative”, then index it as-is, following the UI for capitalization. If, however, it is general, such as “Destination File Name”, “Release”, “Network Directory”, or “Command Line”, then index it in a more general way, using lowercase and according to the user’s goal.

Avoid:	Use this instead:
Automatically add advertising information checkbox	advertising information file adding advertising information
Release drop-down list	administrative install 65 Distribution Method 7, 13, 54 Distribution Wizard 55 floppy disk distributing to FTP server distributing to installation distributing network distributing to release distributing

Verbs in Index Markers

This section contains guidelines for when and how to use verbs in index markers.

See:

[Verb Form for Indexing](#)
[Starting Index Markers with Verbs](#)

Verb Form for Indexing

If the verb is the primary index marker, try to use the noun form of the verb. Sometimes this isn’t possible, or the noun form doesn’t read well (such as with “compilation”). If you can’t use the noun form, use the gerund form (gerunds end in “ing”).

If the verb is a subentry, use the gerund form of the verb.

Avoid:	Use this instead:
advertising	advertisement
compile	compiling

Avoid:**Use this instead:**

compile errors
display
print

compile errors
displaying
printing

Starting Index Markers with Verbs

First, determine whether the verb has a very specific meaning, or whether it is a common, general verb. If it is a common, general verb, such as “using”, do not use it as a primary index marker. Instead of focusing on the verb, focus on the noun that the user is working with. Example: In the index marker “adding files”, the noun is “files”. Therefore, the word “file” should be used as the primary index marker, and “adding” should be a subentry to describe the action taken on the file.

Note

Some verbs, such as “installing”, might have specific meanings in other products, but that specificity is lost in our product.

Second, if it *is* a specific type of verb, determine whether you could use the noun form of the verb instead of the verb itself. Example: Use “advertisement” instead of “advertising”. This doesn’t always work, as in the case of “compilation” and “importation”, which don’t read well.

The examples below illustrate how to convert index markers that use general verbs. Below the examples are lists of suggestions on what kinds of words to use and to avoid as primary index markers. The lists are not meant to be definitive, complete lists; they are provided to give you a general sense of what kinds of verbs to use or avoid.

For the subentries under an index entry, start them with nouns if possible also. See the suggestions below the example.

Avoid:**Use this instead:**

about Visual MSIDiff
adding
files
services
configuring
file attributes
services
editing files
installing files
using Visual MSIDiff

file
adding
configuring attributes
editing
installing
service
adding
configuring
Visual MSIDiff
about
using

file
adding multiple files
adding shortcuts
creating duplicates

file
duplicate, adding
multiple files, adding
shortcut, adding

Suggestions:

Don't Use: adding, configuring, creating, editing, installing, setting, using.

OK to Use: comparing, compiling, exporting, importing

Chapter 3

Wise Terminology

Note

This section will be replaced by the Altiris Terminology document. - JK

Follow the guidelines in this section to maintain consistency in terminology across the various Wise documentation sets.

Some of these terms were taken from Microsoft's *Manual of Style for Technical Publications*. When referring to computer terms, especially Windows terms, Microsoft's standards will prevail unless we have a reason to deviate from them. Otherwise, we follow the standards of *The Chicago Manual of Style*, *Webster's Dictionary*, *The Elements of Style*, and other generally-accepted grammar and style guides.

If you are in doubt about how to format or refer to an item that is not listed here, ask Wise's Technical Publications manager.

To find common usage for a new or unfamiliar computer term, refer to these Web sites:

dictionary.reference.com/
www.techweb.com/encyclopedia/
www.wordspy.com/TechWordSpy/
www.computeruser.com/resources/dictionary/
www.webopedia.com/
www.hyperdictionary.com/computer
www.techdictionary.com/

Terms	Notes
800-456-2100	Format for phone numbers.
&	<ul style="list-style-type: none">◆ Do not use the ampersand character unless you are typing something that appears in the UI. Use <i>and</i> instead.◆ When referring to an ampersand in text, you can use just the symbol. Example: <i>The letter that follows the & becomes underlined.</i>
above, below	Do not use to refer to other topics or illustrations in a manual. The relative position of the referenced items might change and this type of reference creates problems in online help.
accelerator key	Use shortcut key .
action	See script action .
affect, effect	<i>affect</i> (v.): The change in format <i>affects</i> only the selected text. <i>effect</i> (n.): Clicking Apply has no <i>effect</i> . <i>effect</i> (v.): Clicking Apply <i>effects</i> a change. (Avoid this construction.)
all of the	Generally the <i>of</i> is not necessary. Say <i>all files</i> instead of <i>all of the files</i> .

Terms	Notes
allows, lets, permits	Do not use <i>allows</i> . Rewrite the sentence to avoid the idea of permission. Example: <i>Use the Find feature to...</i> rather than <i>The Find feature allows you to...</i> If a rewrite is impossible, use <i>lets</i> . Example: <i>The Find feature lets you...</i> “Allows” implies permission and sounds condescending, while “lets” is enabling. Example: <i>The Call DLL action lets your installer execute a DLL during installation.</i> Or, <i>The Registry page lets you add, delete, and edit registry entries.</i>
and/or	Rewrite to avoid this construction. This has no equivalent in some languages.
appears	Use when there is no object. Examples: <ul style="list-style-type: none"> ◆ The Wise Options dialog appears. ◆ A message appears listing the databases requiring conflict detection. ◆ If you entered a wildcard, it appears in the File Name column. ◆ The Preflight button appears only if Quality Assurance Module - Preflight Server is installed on this computer. <p>Also see displays on page 55.</p>
Application Integration Suite	Obsolete. Product name was changed to Wise Package Studio .
Application Isolation	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
Application Gateway	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
ApplicationWatch	One word. Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
area	When referring to a labeled portion of a dialog, use <i>section</i> instead.
author, authored	Do not use as a verb. Use forms of <i>write</i> or <i>written</i> instead.
Autoexec.bat	Use the initial cap. Some of the product UI might spell it AUTOEXEC.BAT, but use <i>Autoexec.bat</i> consistently in the documentation.
automatically	Generally you can omit this. Instead of <i>the file is added automatically</i> , just say <i>the file is added</i> . Do use this to emphasize that something happens without user action, usually through the selection of an option or use of a special feature. <ul style="list-style-type: none"> ◆ <i>Mark this option to have source paths updated automatically.</i> ◆ <i>If the .NET Framework is installed on your computer, the following tasks are performed automatically when you add a .NET assembly.</i> ◆ <i>You can run a scheduled task that imports packages automatically.</i>
back up (verb)	Two words. Example: <i>Back up your files frequently to prevent data loss.</i>
backup (noun, adj.)	One word, no hyphen. Example: <i>If you need to revert to a backup, follow these steps.</i>
backslash, slash	A backslash is \ and a slash is /. Do not use <i>backward slash</i> or <i>hack mark</i> .
because, since	Use <i>because</i> to refer to a reason; use <i>since</i> to indicate a passage of time.
below	See above , below .
bottom left/bottom right	Don't use; use <i>lower left</i> or <i>lower right</i> as a noun; <i>lower-left</i> or <i>lower-right</i> as an adjective.
box, boxes	In general, do not use. Do not say <i>list box</i> , say <i>list</i> or <i>drop-down list</i> . Do not say <i>text box</i> , say <i>field</i> or <i>static text control</i> or <i>text object</i> .
bring up	Do not use. Instead of <i>Clicking the Settings button brings up the Settings dialog</i> , say <i>Click the Settings button to display the Settings dialog</i> .

Terms	Notes
browse button	You do not need to describe the Browse button at each occurrence. Instead, say <i>In the xxx field, enter yyy, or Specify yyy in the xxx field.</i> In some cases, for clarity, you might mention the ability to browse. Example: <i>You can browse to the .EXE files on your computer and local network, or Specify the .EXE file to launch. The Browse button shows you the .EXE files that you've added to this installation.</i>
bulleted lists	If the items in a bulleted list form a sentence, end them with a period. If not, don't end them with a period.
buttons	<ul style="list-style-type: none"> ◆ Omit "the" and "button" surrounding button names. Example: <i>Click Add Files.</i> Do include "the" and "button" if it's needed for clarity. For details, see Buttons and Button Menus on page 18. ◆ You <i>click</i> buttons. ◆ Do not refer to tools as buttons. If it is on a toolbar, it is a tool, not a button. See tool, toolbar.
button menus	Sometimes clicking a button displays a list of options. Usually, you do not need to mention the button menu. Example: <i>Click Add and select Import.</i> If needed for clarity: <i>Click Add to Project and select a project from the button menu.</i> Do not use <i>button's menu</i> or <i>button's drop-down list</i> .
capitalization	In general, capitalization should be consistent with the user interface. Use initial caps for all heads (all levels). See individual words in this list for specific capitalization information.
checkbox	One word. You <i>mark</i> or <i>clear</i> a checkbox.
choose	Use <i>select</i> instead if you're talking about selecting a menu command from a menu or selecting an option from a drop-down list. In general, for something that you click and drag through that becomes highlighted, say <i>select</i> .
clean machine	In this case, use <i>machine</i> instead of <i>computer</i> because <i>clean machine</i> seems to be the industry term. But relate it back to a "computer". Example: <i>A clean machine is a computer that ...</i>
click	<ul style="list-style-type: none"> ◆ Not <i>click on</i> or <i>press</i>. If the user is doing something with the mouse, they are clicking. Use <i>press</i> only for pressing keys on the keyboard. ◆ You can say <i>click in</i>. Example: <i>Click in the right pane.</i> ◆ Use <i>select</i> for items that become highlighted when you hold the mouse button down over them. Example: <i>Select one or more files to add.</i>
command	Use instead of <i>menu option</i> or <i>menu item</i> . In general, refer to menu commands as in the examples below. <i>Select File menu > New.</i> Capitalize commands exactly as they appear onscreen; if used as a verb, use lowercase (<i>the New command; paste the object...</i>). If an ellipsis appears after a menu command, do not include it in references in text. Example: <i>Select Browse and find the file.</i> Not: <i>Select Browse... and find the file.</i>
Command Line Builder	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
comma	Always use the serial comma (a comma preceding <i>and</i> or <i>or</i> in a list of 3 or more items). Example: <i>You can select cut, copy, or paste from the Edit menu.</i>

Terms	Notes
compile (v., n.)	Even though it is grammatically incorrect, use <i>compile</i> as a noun. Example: <i>During compile, all files are packed into the .EXE file.</i> Grammatically, you should use the word <i>compilation</i> . However, we must remember our audience, and they all say <i>compile</i> to refer to the compilation process. To use <i>compilation</i> in this instance would be unnecessarily confusing to our audience.
component	THIS IS A RESERVED WORD. Do not use this unless you are specifically referring to a <i>component</i> as defined by Microsoft. Microsoft uses <i>component</i> to describe the smallest installable unit.
computer	Not <i>machine</i> or <i>PC</i> or <i>system</i> . (Use <i>system</i> only if you are really talking about some kind of "system.")
Config.sys	Use the initial cap. Some of the product UI might spell it CONFIG.SYS, but use <i>Autoexec.bat</i> consistently in the documentation.
ConflictManager	In general, do not precede the product name with <i>the</i> .
contractions	Contractions are allowed in Wise technical documentation.
Convert Installshield® Professional	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
Convert SMS Installer or WiseScript Installation	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
cross-references	Insert cross-references using FrameMaker's cross-reference function. Do not use references such as <i>earlier in this chapter</i> , <i>in the next section</i> , etc. For details, see Cross-References on page 28.
cursor	Use <i>pointer</i> instead.
desire	See want , wish .
destination computer	Use to refer to something that happens at installation time on the user's computer. Not <i>end user's computer</i> , not <i>its future location</i> , not <i>the user's system</i> or any other variants.
dialog	<ul style="list-style-type: none"> ◆ Say <i>dialog</i>, unless the UI specifically says otherwise. ◆ Refer to a <i>section</i> of a dialog; not an <i>area</i>. ◆ You <i>click</i> a tab in a dialog, you do not <i>select</i> it. ◆ Say <i>on the dialog</i> to refer to a control that appears there or something the user does there. Example: <i>... on the File Details dialog...</i> ◆ Dialogs contain <i>options</i> or <i>settings</i>, not <i>controls</i> (except when you're specifically talking about the user adding controls to an installation dialog).
dimmed	Use <i>disabled</i> instead.
directory	Generally, use instead of <i>folder</i> . For exceptions, see folder on page 57.
disabled	Use to describe controls that are unavailable. Do not use <i>grayed-out</i> or <i>dimmed</i> .

Terms	Notes
displays	<p>Use when there is an object. Examples:</p> <ul style="list-style-type: none"> ◆ The groups pane displays the groups defined for each database. ◆ The Description tab in the right pane displays help text. ◆ The URL field displays the path to the Web application. <p>Also use <i>displays</i> with passive voice. In some instances you can probably use <i>appears</i> instead and make it active, but in other cases passive is appropriate. Examples:</p> <ul style="list-style-type: none"> ◆ The new rule is displayed in the upper list box. ◆ When this task is displayed on the Projects tab, clicking its Run link launches the application. <p>Do not use <i>displays</i> when there is no object. Example: <i>The Project Setup dialog displays</i>. Use appears instead.</p>
Distribution Wizard	Do precede with <i>the</i> .
double-click	Always hyphenate.
drag-and-drop	Hyphenated, one word. Do not use as a verb; instead, use <i>drag</i> or <i>dragging</i> .
drop-down list	<p>Not <i>drop-down menu</i> or <i>pull-down menu</i> or any other usage. Users <i>select XYZ from the drop-down list</i>.</p> <p>A drop-down list is a control in a dialog or window. It looks like a field with a little button to the right that has a down arrow on it. If you click the down arrow, you see a list of options, which might or might not have a scroll bar. You select an option from a drop-down list. Example: <i>Select Windows 98 from the Windows Version drop-down list</i>.</p>
drop-down menu	Use <i>drop-down list</i> instead.
earlier	Use to refer to product version numbers. Example: <i>Wise Package Studio version 5.0 or earlier</i> .
e.g.	Don't use. Use example . Also see Introducing Examples on page 21.
edit box	Use <i>field</i> instead. Do not use <i>editable text box</i> .

Terms	Notes
edition, module	<p>When describing different editions of the product, lowercase the word edition. It is part of the full product name, but we rarely use it that way in the documentation. In most cases, do not use the word edition at all: <i>Wise Installation System Standard</i>, <i>Wise Installation System Professional</i>.</p> <p>In caveats at the beginning of a topic, omit the article.</p> <ul style="list-style-type: none"> ◆ Professional Edition only. ◆ Enterprise Management Server only. ◆ Quality Assurance module only. ◆ Application Gateway module only. <p>In parentheses before or within a sentence, omit the article.</p> <ul style="list-style-type: none"> ◆ (Enterprise Management Server only.) To duplicate a process template from an external corporate database, open the external database. [Include the period because the text in parentheses is a standalone sentence.] ◆ Use the Projects tab to work on a process-oriented project (Professional Edition only). <p>In a sentence, use articles for the editions and the modules except Enterprise Management Server.</p> <ul style="list-style-type: none"> ◆ In the Standard Edition, the Finish dialog appears. ◆ In the Professional Edition, the ApplicationWatch Inclusions dialog appears. ◆ With the Quality Assurance module, you can customize validation modules. ◆ With the Application Gateway module, the Requests tab appears. ◆ With Enterprise Management Server, each project has an owner. [Sounds better without the article.]
effect, affect	See affect , effect .
either/or	<ul style="list-style-type: none"> ◆ Do not use <i>either/or</i>. be specific. ◆ Use <i>either</i> and <i>or</i> only for a list of 2 items. Do not use <i>either</i> to introduce a series of 3 or more choices. Example: Do not use <i>either</i> like this: <i>This option copies the file to either the Windows dir, the Program Files dir, or the My Documents dir</i>. That is incorrect. For a series of 3 or more, omit <i>either</i>.
ellipsis	If an ellipsis appears after a menu command, do not include it in references in text. Example: <i>Select Browse and find the file</i> . Not: <i>Select Browse... and find the file</i> .
e-mail	Use the hyphen. Do not use <i>email</i> . Most style guides, including <i>Chicago Manual of Style</i> and Microsoft, are adamant about using the hyphen.
enabled	Use to describe available controls.
end user	<p>No hyphen. The person reading our documentation and using our software is referred to as <i>you</i> or, less often, <i>the user</i>. The reader's customer is referred to as the <i>end user</i>. The Wise Package Studio UI refers to <i>users</i>, which are our customers. In Package Studio it's especially important to distinguish between the <i>user</i> and the <i>end user</i>.</p> <p>Do not say <i>end user interface</i>.</p>
equals sign	<ul style="list-style-type: none"> ◆ Not equal sign. Example: <i>Place the full pathname within quotation marks after the equals sign</i>. ◆ Even better: use the symbol. Example: <i>Place the full pathname within quotation marks after the =</i>.
et. al.	Use <i>and others</i> instead.

Terms	Notes
etc.	Use <i>and so on</i> instead.
example	Use <i>example</i> or <i>examples</i> by itself to introduce examples. Not <i>for example</i> , or <i>for instance</i> , or <i>here is an example</i> . See Introducing Examples on page 21 for formatting guidelines. Cultural considerations in examples: Keep sample text (example: fictitious company or product names) generic. Widget, gadget, and other such sample names will not translate well. Instead, use the following: <ul style="list-style-type: none"> ◆ Application Names: Application1, Application2, Application3, etc. ◆ Packages: Package1, Package2, Package3, etc. ◆ Company Names: Company1, Company2, Company3, etc. ◆ Various: Sample1, Sample2, Sample3, etc.
field	Use <i>field</i> instead of <i>edit box</i> or <i>text box</i> to indicate where a user types.
field entries	When telling the user how to make an entry in a field, say <i>In xxx, enter yyy</i> , or <i>In the xxx field, specify yyy</i> . Use this even when the field includes a Browse button. In some cases, for clarity, you might need to mention the ability to browse. Example: <i>You can browse to the .EXE files on your computer and local network.</i>
file extensions	If you are talking about a certain kind of file in general, capitalize and include a period before the extension. Do not use the period if the file extension begins a sentence or heading. Select the article before a file extension based on how it is pronounced. Example: <i>an .EXE, a .DLL</i> . When forming the plural of file extensions, add an s (.EXEs, .DLLs). If that doesn't work, then change the wording to include the word "file" and pluralize the word "file". Example: <i>If you have .MSI files that...</i> If you are talking about a particular file, type it as its name appears in Windows Explorer. Example: <i>Add the following text to the system.ini file.</i> If you are talking about a certain kind of file, and you want to mention the file extension to make it clear, put the extension in parentheses, in uppercase, but with a period. Example: <i>To create a Windows Installer database file (.MSI), click Compile.</i>
file name, pathname	File name is 2 words. Pathname is 1 word. This is because our UI generally displays it as all one word. Use <i>pathname</i> when you refer to an entire path, such as <i>C:\program files\wise\wise.exe</i> , and use <i>file name</i> when you refer to just the file name, such as <i>Enter the file name, such as wise.exe, in this field.</i>
fill in, fill out	Use <i>complete</i> instead. <i>Complete the dialog and click OK</i> . Or, when introducing a series of controls in a procedure: <i>Complete the x dialog:</i>
first person	Don't use; rewrite. First person is <i>We recommend...</i> or <i>I recommend</i> .
folder	Generally, use <i>directory</i> instead. Use <i>folder</i> only when referring to the graphical representation of a folder (example: <i>the folder icon</i>). If it is in the UI, get the UI changed if possible. Don't write bugs on "folder" in button names (example: New Folder).
gray, grey	Use <i>gray</i> (both spellings are correct).
grayed out	Use <i>disabled</i> instead.
higher	Use to refer to more powerful hardware. Example: <i>a 386 or higher</i> . Do not use to refer to product version numbers. Use <i>later</i> instead. Example: <i>Wise Package Studio 5.0 or later</i> .
hit	You do not <i>hit</i> a key on the keyboard, you <i>press</i> it.

Terms	Notes
hold down, press	If you want the user to really hold down a key, then say <i>hold down</i> . Example: <i>To preview your installation, hold down the Shift key while you click the Preview button</i> . If you just mean press, then say <i>press</i> . Example: <i>Press Enter to go to the next line</i> . If they press a combination of keys, just say <i>Press Ctrl+F to find specified text</i> . Conversely, you do not <i>press</i> a button onscreen. You <i>click</i> buttons onscreen.
i.e.	Use <i>that is</i> instead.
if you want to	Just say <i>to</i> ; the other words are not necessary. In rare cases, you might use this to indicate that something is optional but in those cases, try using (Optional) instead.
Import SMS/WiseScript	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
Import VB Project	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
in order to	Use <i>to</i> instead.
install	This is a verb. Do not use it as a noun. Example: <i>When the user installs your application, ...</i> Not: <i>If your install contains a Call DLL action, ...</i>
installation	<ul style="list-style-type: none"> ◆ This is the .WSE file in the Wise Installation System or the .WSI file in WfWI. Example: <i>You use Wise Installation System to build an installation.</i> ◆ As an adjective: <i>You must edit your installation script to include custom dialogs.</i> ◆ As the process that happens on the destination computer: <i>During installation, files are copied to the destination computer.</i>
installation-on-demand	Use the hyphens to refer to this Microsoft Windows Installer feature. Microsoft refers to it with hyphens, not without. (Incorrect: installation on demand)
Installation Expert	Capitalize to be consistent with UI. Do not use <i>Expert</i> . Do not precede with <i>the</i> .
installer, setup program	Use <i>setup program</i> . This is the .EXE generated by the Wise Installation System. It is a file that installs software. You launch or run an installer to put new software on your computer. It is the file that is compiled by a .WSE file.
InstallTailor	One word. Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
Internet	
Internet Information Server	Use Microsoft Internet Information Server, NOT Microsoft Internet Information Services. Here are the reasons: <ul style="list-style-type: none"> ◆ Microsoft uses both "services" and "server", seemingly interchangeably, on their own Web site. ◆ If you type each entire phrase in google enclosed in quotes, the "Services" phrase gets ~ 22,000 hits, while "Server" gets over 160,000 hits. ◆ I suspect that "server" refers to the "full" version and "services" refers to the "personal version". I suspect that the "full" version is called Internet Information Server.
key path	Not "keypath" or "KeyPath". Only use it as one word when referring to something in UI, such as "the KeyPath field", "The KeyPath column". The Windows SDK is not consistent, but seems to use "key path" the most.

Terms	Notes
later	Use to refer to product version numbers. Example: <i>Wise Package Studio 5.0 or later</i> . Do not use <i>later</i> or <i>previously</i> to refer to upcoming or past material in the documentation. The relative position of the referenced items might change and this type of reference creates problems in online help. Also, users use reference manuals to look up specific topics—they don't usually read them like a novel, from front to back.
left-hand	Just say <i>left</i> .
Legacy Setup Conversion	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
let, lets	See <i>allows/lets/permits</i> .
list box	Use only when referring to generic or unnamed list boxes. Example: <i>Using the upper two list boxes...</i> , <i>In the upper left list box...</i> When referring to a named list box, use <i>list</i> . Example: <i>the Source Directory list</i> .
locations on screen	Examples describing positions relative to the edges of the screen (or the page): <i>top of the window</i> <i>bottom of the window</i> <i>left side of the window (or on the left)</i> <i>right side of the window (or on the right)</i> <i>lower left, lower right</i> <i>upper left, upper right</i> Examples describing positions relative to other objects (on one plane): <i>above</i> (not over) <i>below</i> (not under or beneath) Examples describing positions relative to other objects (on several planes): <i>in front of</i> (not over) <i>below</i> (not under or beneath)
log on, logon	Use: <ul style="list-style-type: none"> ◆ Logon. Use as an adjective to describe the information you use to log on. Example: <i>Your logon information is usually your NT name and password.</i> ◆ Log on. Use as a verb to describe what the user does to gain access to a computer, network, etc. Example: <i>When Package Studio opens, log on with your name and password.</i> ◆ Log off. Example: <i>You log off your computer when you are done with it.</i> ◆ Logged on. Use as a noun to describe the state of the user, OR, use as a past tense verb, etc. Example: <i>The user is currently logged on.</i> OR <i>He logged on, and the network immediately went down.</i> ◆ Logging on. Example: <i>This survey had to do with logging on to servers.</i> Do Not Use: login, log in, logged in, logging in
lower	Do not use to refer to product version numbers. Use <i>earlier</i> instead. Example: <i>Wise Package Studio 5.0 or earlier</i> .
lower left, lower right (n.)	Don't use <i>lower left-hand, lower right-hand, bottom left, or bottom right</i> .
lowercase	One word.
lower-left, lower-right (adj.)	Note hyphenation of adjective.
machine	Use <i>computer</i> instead, except when referring to a clean machine .

Terms	Notes
may	Do not use this word at all. Used in this way, <i>He may go to school now</i> , it can mean either <i>He has permission to go to school now</i> , or <i>He might go to school now</i> . Because of this built-in ambiguity in the word <i>may</i> do not use it. Use either <i>can</i> (for permission situations) or <i>might</i> (for probability) instead. Also, <i>may</i> used as permission sounds condescending, while <i>can</i> sounds enabling.
media, medium	Use <i>media</i> in all cases, even as a singular subject. Make the verb agree with the subject in the context of the sentence. Example: <i>Verify that the media is in the drive</i> , and <i>Various types of media are supported</i> .
menu option, menu item	See command .
menus	A menu is at the top of a window. You <i>select</i> a <i>command</i> from a menu. Example: <i>Select Edit menu > Preferences</i> . Do not capitalize the word <i>menu</i> when used after a menu name (<i>Edit menu</i>). See Menus on page 19.
menus on buttons	See button menus .
meta data	Two words, no hyphen. Although the term <i>metadata</i> , spelled the same way, is widely used to refer to <i>data about data</i> , Metadata trademarked the name in 1986 and was granted “incontestable” status in 1991. Per TechEncyclopedia (www.techweb.com/encyclopedia/).
Microsoft	Do not abbreviate as MS.
mobile device	Use to generically describe handheld computing devices such as Pocket PC, Smartphone, or Palm.
Mobile Device Package Editor	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
modify	Use <i>edit</i> instead.
module	See edition , module .
MSI Script	Do not precede with <i>the</i> except when referring to <i>the MSI Script tab</i> .
.NET	Per Microsoft, always use upper case, always use the period. Never use <i>dot NET</i> . In first mention, use the full name: <i>Microsoft® .NET</i> .
need to	Use <i>must</i> instead. In many cases it is unnecessary.
newsgroups	One word.
numerals	Per Abreve, use numerals in text, even for numbers 1 through 9. Exceptions: <ul style="list-style-type: none"> ◆ Spell out numbers that begin a sentence. ◆ Spell out one when it doesn’t specifically refer to a number. Example: <i>select the one that appears...</i>
online	One word.
onscreen	One word.
option	Use to describe a radio button selection. Example: <i>Mark the Windows 2000 option</i> .
Package Studio	See Wise Package Studio .
PackageManager	Obsolete. Product name was changed to Wise Package Studio .
Package Distribution	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
Package Validation	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .

Terms	Notes
page	Use to describe each of the pages you can go to in Installation Expert. Example: <i>Use the Files page to specify the files to include in the installation.</i> Say <i>on the page</i> to refer to a control that appears there or something the user does there. Example: <i>... on the Registry page.</i>
parentheses	Do not use to enclose phrases that are meant to add to or clarify information. Do not enclose cross-refs in parentheses. Use sparingly if at all. If parentheses can be replaced with commas and not change meaning, replace them with commas. If parentheses can be replaced with an em dash (—) do so.
passive voice	Do not use unless absolutely necessary. Use active voice.
Patch Creation	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
pathname, file name	See file name , pathname .
PC	Use <i>computer</i> instead, unless you are talking about a specific type of computer
permit, permits	See allows , lets , permits .
point-and-click	Hyphenated, one word.
pointer	Not <i>cursor</i> .
pop-up menu	Do not use when referring to the result of a right-click; use <i>right-click menu</i> instead.
predefined	Not <i>pre-defined</i> .
Preflight Analysis	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
Preflight database	
Preflight Instrumentation	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
press/hold down	You <i>press</i> a key on the keyboard. You do not <i>hold down</i> a key. Conversely, you do not <i>press</i> a button onscreen. You <i>click</i> buttons onscreen.
previously	Do not use <i>later</i> or <i>previously</i> to refer to upcoming or past material in the book. The relative position of the referenced items might change and this type of reference also creates problems when the manual is converted to online help. Also, users use reference manuals to look up specific topics—they don't usually read them like a novel, from front to back.
Process Setup	Describes a Wise Package Studio function. In general, do not precede with <i>the</i> .
Project Setup	Describes a Wise Package Studio function. In general, do not precede with <i>the</i> .
Projects tab	One of the tabs in Wise Package Studio's Workbench. DO precede with <i>the</i> . Example: <i>Open a project on the Projects tab.</i>
pull-down menu	Do not use. Use <i>menu</i> , or use <i>drop-down list</i> if appropriate.
punctuation	Use 1 space between words, and 1 space between final punctuation and the next sentence. In Frame, turn on "smart spaces" by selecting Document > Text Options from the Format menu.
Quality Assurance	The official name is <i>Wise Package Studio - Quality Assurance</i> . This is used rarely, usually in introductory material. ◆ In the Wise Package Studio documentation, use <i>Quality Assurance module</i> . ◆ In the QA standalone documentation, use <i>Quality Assurance</i> .
quick-access key	Use shortcut key .

Terms	Notes
radio button	Use <i>option</i> instead: Example: <i>Mark the Windows 2000 option</i> . A radio button option is <i>marked</i> or <i>cleared</i> . DO say <i>radio button</i> when you're describing how to add or edit radio button controls on a dialog.
ReadMe	Capitalize the R and the M. Use this capitalization for all instances of ReadMe, whether you are talking about ReadMe files in general, or the ReadMe file in the Wise installation directory.
registry	Use <i>registry</i> . Do not say <i>Windows Registry</i> or <i>Registry</i> . Microsoft refers to it as <i>the registry</i> .
re-run	Use to mean "run the program again." Example: <i>Re-run the installer</i> .
right-click menu	Use instead of <i>shortcut menu</i> or <i>popup menu</i> .
runtime	Use 1 word in all instances.
sample text	See example on page 57.
screen shots	Do not refer to screen shots in the text above or below the shot because many screen shots are removed during the conversion to help. See Creating Screen Shots and Placing Graphics on page 100.
script action	Use to describe the items in the Actions list in Script Editor or MSI Script. In general, to prevent ambiguity do not use <i>action</i> by itself (the user could assume the dictionary meaning of action, instead of relating it to the list of Actions in Script Editor.) However, after you first use <i>script action</i> in a paragraph you can shorten it to <i>action</i> for the rest of the current paragraph. When discussing specific actions, you usually can omit <i>action</i> . Example: <i>In the Actions list, double-click Call Custom DLL from Destination</i> . Not: <i>In the Actions list, double-click the Call Custom DLL from Destination script action</i> .
Script Editor	Capitalize for consistency with UI. Do not say <i>Editor</i> . Do not precede with <i>the</i> .
script line	Use to describe a line in a script. A script line is made up of a script action plus the options the user has set for the script action. After you first use <i>script line</i> in a paragraph you can shorten it to <i>line</i> for the rest of the current paragraph. When <i>line</i> is used in conjunction with an action, use <i>action</i> instead. Example: <i>Above the Execute Program From Destination action...</i>
select	Use for: <ul style="list-style-type: none"> ◆ Menu commands: <i>Select File menu > New</i>. ◆ Button menu commands: <i>Click the Color button and select a new color from the Color Picker</i>. ◆ Options from drop-down lists or list boxes: <i>Select x from the y drop-down list</i>. ◆ Items that become highlighted when you hold the mouse button down over them: <i>Select one or more files to add</i>. ◆ General selecting: <i>Files are added to the directory you selected. On the Copy Source Files dialog, select the file to copy</i>.
self-register (v.) self-registration (n.)	Hyphenate.
semicolon	One word.

Terms	Notes
sentence structure	ACTIVE VOICE Second person Present tense Singular number Correct: <i>When you click the Show Tables button, the product displays...</i> <i>Use the Display option to let users see tables.</i> Incorrect: <i>When the Show Tables button is clicked, the product will display...</i> <i>We recommend that you use the Display option because then users can see the tables.</i>
serial comma	See comma .
setup program	See installer , setup program .
SetupCapture	One word. Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
share point, share point directory	<i>Share point directory</i> is preferred. Use share point directory to refer to the actual directory structure. Use the generic term instead of Wise Share Point; users can change the name of the directory during installation, so it might not be called that on their computers. For marketing purposes you can use <i>Wise Share Point</i> .
shortcut key	A keyboard combination that invokes a command. Typically, we use it to refer to adding an underline to a character in a menu or other command, to indicate which character in the command is used with a shortcut key. Do not use <i>accelerator key</i> or <i>quick-access key</i> .
shortcut menu	Use <i>right-click menu</i> instead.
since	See because , since .
slash/backslash	A backslash is \ and a slash is /. Do not use <i>backward slash</i> or <i>hack mark</i> .
Smartphone	One word
SOE Snapshot	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
Software Manager	Two words. Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
Software Manager database	Previously referred to as the <i>Wise Software Repository database</i> or <i>repository database</i> . Those terms are obsolete.
spaces	Use 1 space between words, and 1 space between final punctuation and the next sentence. In Frame, turn on "smart spaces" by selecting Document > Text Options from the Format menu.
stand-alone	Hyphenate.
step	Use to describe the steps as in <i>the files and components step</i> . Example: <i>Click the System Additions step, then click INI Files</i> . Obsolete as of WfWI 3.5 and Wise Installation System 9.0.
step button	Example: <i>The pages you select appear on the step button</i> . Obsolete as of WfWI 3.5 and Wise Installation System 9.0.
subdirectory	Do not hyphenate.

Terms	Notes
symbols	<p>When referring to a symbol in text, you can use just the symbol, unless it would be unclear. Examples: <i>The letter that follows the & becomes underlined.</i> <i>Enclose the property in [].</i> <i>Type a \ after the file name.</i></p> <p>Select the article before the symbol based on how it is pronounced. <i>An &, a ?, an *</i>.</p>
tab	<p>In general, do not capitalize <i>tab</i> unless it is capitalized in the UI. Example: <i>Installation Expert tab, Tables tab.</i> <i>Say on the tab</i> to refer to a control that appears there or something the user does there. Example: <i>Select a table on the Tables tab.</i></p>
Test Expert	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
that, which	<p><i>That</i> introduces a restrictive clause. A restrictive clause is a clause that is essential for the sentence to make sense. It is not separated from the noun or phrase that precedes it by a comma. <i>Which</i> introduces a nonrestrictive clause, which is a clause that could be omitted without affecting the meaning of the sentence. A comma always precedes it. For examples of correct usage of <i>that</i> and <i>which</i>, see their respective descriptions above.</p>
there is, there are	Rewrite to avoid this construction.
title bar	Two words. This is the colored bar at the top of a window or dialog that contains the name of the current window.
tool names	<p>Do not precede with <i>the</i>; do not append <i>tool</i> or <i>wizard</i>.</p> <ul style="list-style-type: none"> ◆ There might be times when <i>the <whatever> tool</i> is needed for clarity. ◆ You can use <i>tool</i> by itself for brevity, rather than repeating the tool name. Example: <i>ApplicationWatch does this, that, and the other. Use this tool to...</i>
Tool Setup	Describes a Wise Package Studio function. In general, do not precede with <i>the</i> .
tool, toolbar	You can find <i>tools</i> on <i>toolbars</i> . You <i>click</i> a tool, you do not choose it. Example: <i>Click the Rectangle tool on the Custom Billboard toolbar and then drag a box to create a new rectangle.</i>
Tools tab	One of the tabs in Wise Package Studio's Workbench. DO precede with <i>the</i> . Example: <i>Launch ConflictManager from the Tools tab.</i>
tooltip	One word. These are the little flyout labels that appear if you hold your pointer over tool icons.
UpgradeSync	One word. Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
uppercase	One word.
upper-left, upper-right (adj.) upper left, upper right (n.)	Note hyphenation of adjective. Don't use <i>upper left-hand, upper right-hand, top left, or top right</i> .

Terms	Notes
URLs	Do not include URLs in documentation unless they refer to the top level of a Web site, such as www.microsoft.com or www.wisesolutions.com . Never include URLs that refer to deeply nested HTML files on a Web site. This is for 2 reasons: because it is easy for the user to mis-type a long URL, and because URLs change so frequently that they do not belong in a static communication medium, such as a book that might be around for up to a year. Instead of giving the user an URL, say <i>For information on code signing, search for "open software description" in the MSDN Library (msdn.microsoft.com/library/)</i> . You can format URLs so that they become hyperlinks in the help; see Creating Links to URLs on page 106.
user	See end user .
user-defined	Hyphenated. Example: <i>user-defined actions</i> .
user name	Two words. When possible, get the UI changed to <i>User Name</i> , not <i>Username</i> .
utilize	Use <i>use</i> instead.
verbs	Use simple present tense whenever possible. Avoid the passive voice, except where the content clearly calls for an exception. Use the imperative for procedures and cross-references.
via	Do not use. Avoid substituting <i>by way of</i> . Rewrite.
Visual MSIDiff	Do not precede with <i>the</i> ; do not append <i>tool</i> or <i>wizard</i> .
Visual Studio .NET	On first reference, use <i>Microsoft® Visual Studio® .NET</i> . Thereafter, <i>Visual Studio .NET</i> is OK. The <i>.NET</i> is always in uppercase, and there is always a space between <i>Visual Studio</i> and <i>.NET</i> .
Visual Studio integrated editor	In Wise for Windows Installer, the interface that is integrated with Visual Studio .NET. Before version 6.0, this was a separate product named Wise for Visual Studio .NET. See Documenting the Visual Studio Integrated Editor on page 137.
want, wish	Try not to use either, per Abreve. Otherwise, use <i>want</i> .
Watch Application wizard	Has been renamed <i>ApplicationWatch</i> .
we	Don't use first person.
Web	Not <i>web</i> or <i>website</i> . Likewise <i>Web server</i> , <i>Web page</i> , <i>Web site</i> .
which	See that, which .
Windows	Do not abbreviate as "Win". Also see <i>Windows versions</i> .
Windows 2000 Verification Wizard	Has been renamed <i>Package Validation</i> .
Windows Installer Editor	Do not precede with <i>the</i> . Do not use <i>WIE</i> .
Windows Installer SDK Help	Do precede with <i>the</i> .

Terms	Notes
Windows versions	Always include the word <i>Windows</i> . Do not use <i>Win</i> , as in <i>Win95</i> . Windows 95 Windows 98 Windows 9x—collectively refers to 95, 98, and Me Windows Me—use <i>Windows Me (Millennium Edition)</i> in the first reference; use <i>Windows Me</i> thereafter. Note the lowercase e. Windows NT Windows 2000—do NOT use <i>Win2K</i> . Windows XP
WinINSTALL	The full product name is Veritas WinINSTALL™. In documentation for Package Studio’s WinINSTALL Import Wizard, <i>WinINSTALL</i> is sufficient.
WinINSTALL Import Wizard	Do precede with <i>the</i> . Example: <i>Run the WinINSTALL Import Wizard.</i>
Wise	Do not use by itself. It’s not clear if it means <i>Wise Installation System</i> , <i>Wise for Windows Installer</i> , or <i>Wise Solutions</i> . Exception: To avoid using the product name in a single-sourced book. Example: ... <i>for backward compatibility with installations created in previous versions of this Wise product.</i>
Wise editor	The stand-alone interface of Wise for Windows Installer (as opposed to the interface that is integrated in Visual Studio .NET).
Wise for Visual Studio .NET	Obsolete. As of version 6.0, this is included in the Wise for Windows Installer product and called <i>Visual Studio integrated editor</i> . See Documenting the Visual Studio Integrated Editor on page 137.
Wise for Windows Installer	Do not use <i>WFWI</i> .
Wise Installation System	Formerly called InstallMaker/Builder/Master/Manager. Do not use <i>Wise</i> to refer to this product, because Wise has several products, and <i>Wise</i> could mean the name of the company. Precede with <i>the</i> except when referring to a specific edition. Example: <i>This feature is not available in Wise Installation System Standard.</i>
Wise InstallBuilder Wise InstallMaker Wise InstallManager Wise InstallMaster	Obsolete. Product has been renamed Wise Installation System , Professional Edition and Standard Edition.
Wise Package Studio	Always include <i>Wise</i> . When a paragraph contains multiple instances of the product name, include the <i>Wise</i> in the first instance and you can omit it in subsequent instances. Do not precede with <i>the</i> .
Wise Solutions, Wise Solutions, Inc.	The name of the company.
WiseScript	Describes the Wise scripting language and the actual script that is created in the Wise Installation System. It does NOT describe the product used to create the script. In the first instance of describing the language, say <i>the Wise Scripting Language (WiseScript)</i> .
Wise Software Repository	Describes a concept more than any specific part of our product. <i>Wise Software Repository</i> collectively refers to the information in the Workbench database, the Software Manager database, the Preflight database, and the Wise Share Point. Use <i>Wise Software Repository</i> on first mention. For subsequent mentions in the same paragraph or topic, you can use <i>repository</i> .

Terms	Notes
Wise Software Repository database	Obsolete. Has been renamed <i>Software Manager database</i> .
Wise Software Repository share point	Do not use; there is no such thing. This used to be in the product UI but has been changed. Instead: Package Distribution distributes packages to the share point directory, for later importing to the Wise Software Repository. Software Manager imports packages from the share point directory to the Wise Software Repository.
WiseScript Editor	Do not precede with <i>the</i> . Do not use <i>WSE</i> .
WiseScript Express	Do not precede with <i>the</i> .
WiseUpdate Client	Capitalize both words.
Wise Share Point	Use the generic term <i>share point directory</i> instead; see share point directory .
WiseUser	
wish, want	Try not to use either, per Abreve. Otherwise, use <i>want</i> .
wizard	Do not capitalize the word <i>wizard</i> unless it is part of the full formal name of a wizard, such as <i>Patch Wizard</i> . If you are just saying <i>the wizard</i> , do not capitalize. The formal names of most wizards, such as SetupCapture and InstallTailor, do not include the word <i>wizard</i> .
workaround (n.), work around (v.)	Examples: Verb: Use these steps to <i>work around</i> the problem of the crash. Noun: This article provides a <i>workaround</i> for the crash problem.
Workbench	Wise Package Studio's main interface. In general, do not precede with <i>the</i> . If needed for clarity, you can use <i>Wise Package Studio's Workbench</i> .
Workbench database	

Chapter 4

Tools and Templates

This section discusses the authoring tools we use to produce Wise documentation and how to set them up, as well as the standard templates we use in FrameMaker and WebWorks Publisher.

Topics include:

- ◆ [Tools for Creating Wise Documentation.](#)
- ◆ [FrameMaker Considerations.](#)
- ◆ [Setting Up FrameMaker.](#)
- ◆ [About Frame and Help Templates.](#)
- ◆ [Guidelines for Frame Template Conversions.](#)
- ◆ About the [A4 Page Size.](#)

[Copyright and trademarks.](#)

Tools for Creating Wise Documentation

We prefer to use PCs to produce Wise documentation. Technically, it should be possible to produce Wise documentation on a Macintosh, if all writers use the same version of software tools. However, we have not verified that lately. (See [FrameMaker Considerations](#) on page 71.)

Tool / Version	How It's Used
Adobe FrameMaker 7.0	To create printed and online reference manuals and other documentation.
Adobe Acrobat 6.0	To create .PDF files from FrameMaker files for online reference documentation.
Adobe Generic PostScript driver	To create PostScript files from FrameMaker files for commercial printing of documentation.
Adobe PageMill 3.0	HTML editor for creating Release Notes files. DO NOT use for WfWI inline help pages.
Paint Shop Pro	To capture screen shots and edit bitmap graphics.
RoboHelp Office X3	To create online Help files from FrameMaker documentation files. NO LONGER USED as of 4/03.
WebWorks AutoMap 2003	To automate the help builds.
WebWorks Publisher 2003	To create online help from FrameMaker documentation files.

FrameMaker Plug-Ins

These plug-ins are in Technical Publications\Admin\Documentation Software\Fram Plug-ins.

Plug-In	How It's Used	Available From
Character Tools	Lists character tag usage, removes unused character tags, re-maps character tags, and locates and removes character tag overrides. Also generates a list that defines all character tags. This is the primary tool we use for maintaining templates.	Silicon Prairie Software (www.siliconprairiesoftware.com) \$10 per license.
Paragraph Tools	Lists paragraph tag usage, removes unused paragraph tags, re-maps paragraph tags, and locates and removes paragraph tag overrides. Also generates a list that defines all paragraph tags. This is the primary tool we use for maintaining templates.	Silicon Prairie Software (www.siliconprairiesoftware.com) \$10 per license. We have purchased one license.

Plug-In	How It's Used	Available From
Toolbox Basic Version	Lets you quickly delete unused formats for an entire book. Deletes unused paragraph tags, character tags, user defined variables, cross-reference formats, and table formats.	Systemec (www.systemec-gmbh.com/en/sites/downloads_tk.htm) Freeware.
CudSpan Doc Tools - Hunt Overrides	Finds paragraph overrides and lists them in a new Frame document with hotlinks to the paragraph where the override occurs. We use this mainly to find and fix paragraph tag overrides.	CudSpan (www.telecable.es/personales/cud/) Freeware.
CudSpan Doc Tools - MarkerWorker	Generates a list to represent all markers of a specified type in a book or document. After you generate the list, you can modify the marker text and then rebuild all the listed markers with the new marker text.	CudSpan (www.telecable.es/personales/cud/) Freeware.
BookInfo	Counts the total number of words, pages, and anchored frames in all documents in a book. (Useful when Abreving.)	Bruce Foster (http://www.pacifier.com/~bfoster/) Shareware, \$10 per license.

FrameMaker Considerations

Mac to PC Conversion

This issue was more important when we used outside contractors who worked on Macs. Currently, we only use PCs and are not likely to hire contractors who use Macs. Just be aware of this issue.

It is very important to select the **Remember Missing Font Names** option on FrameMaker's Preferences dialog. If the book files are opened on a system with different font definitions (example: a Macintosh), the paragraph tags defined as Arial are changed to Helvetica, and that change is saved with the document. Then, when the book files are opened at Wise, which uses PCs, all the Helvetica fonts are automatically changed to TimesRoman, which makes it necessary to redefine all the paragraph tags that are supposed to be Arial, which takes 1 to 2 hours for a typical book.

Hyphenation

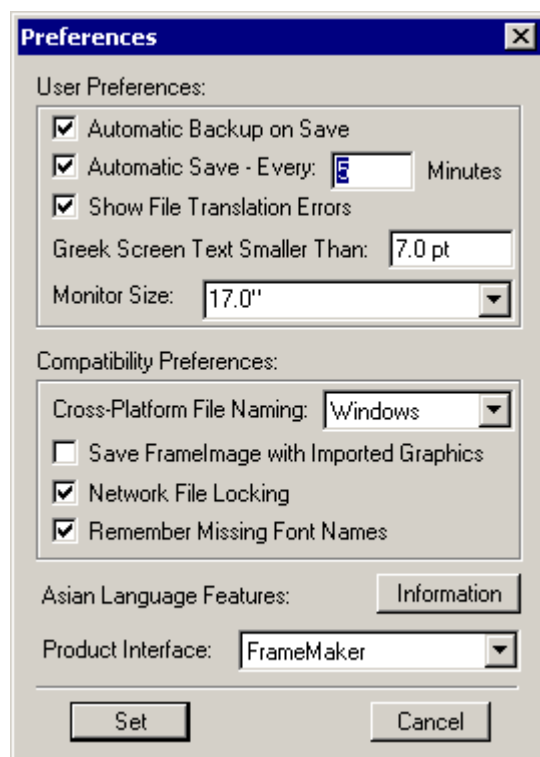
Do not turn hyphenation on in Frame. This used to cause problems in WinHelp. It might not be a problem in HTML help, however, it will cause problems for future localization.

Copying and Pasting From Word

Do not copy and paste rich text from Microsoft Word to Frame. Doing so produces unexpected and inconsistent results, such as multiple hidden tabs. If you must copy and paste from Word to Frame (or vice-versa), do a Paste Special, paste plain text, and reformat it.

Setting Up FrameMaker

1. If the Choose Interface dialog appears when you first launch Frame, click FrameMaker.
2. Select File menu > Preferences > General and set the following preferences:



To specify the Frame personal dictionary:

1. If you are re-installing Frame, and you have a backup copy of your personal dictionary, copy it to the appropriate directory. Typically, this will be one of the following locations:
 - Program Files\Adobe\Framemaker7.0\dict
 - Documents and Settings\
2. Open a document in Frame and open the Spelling Checker dialog.
3. Click Dictionaries.
4. On the Dictionary Functions dialog, mark **Personal Dictionary**.
5. In the drop-down list to the right, select **Change Dictionary** and click OK.
6. On the Use File for Personal Dictionary dialog:
 - Browse to the directory where you want to store your dictionary.
 - Specify the file name (usually user.dct). If you copied a backup dictionary to this location, select it.
 - Click Use.
7. Close the Spelling Checker dialog.

Using Smart Spaces and Smart Quotes

Use only one space between sentences and use typographer’s (“curly”) quotation marks and apostrophes. These features are controlled by the Smart Spaces and Smart Quotes settings in Frame. They are set by document, not globally. To verify that these features are turned on:

1. Open the book or document in Frame and make it active.
2. Select Format menu > Document > Text Options.
3. Mark **Smart Spaces** and **Smart Quotes** option and click Apply.

When you select Smart Quotes, FrameMaker doesn’t change any quotation marks that are already in the document. To find and change straight quotation marks and apostrophes, use the Spelling Checker or the Find/Change command.

About Frame and Help Templates

Technical Publications maintains templates to control the format of books and help systems, listed below. Whenever you add a new style to the documentation template, add a corresponding style to the help templates.

Changes to any of these templates must be approved by the Technical Publications Manager. The template for training materials is based on the documentation template, so discuss any changes with the Curriculum Developer also.

Active Frame Templates

Also see [Differences Between Frame Templates](#) on page 76.

Template Name	Description	Location in Technical Publications
8.5x11 Template.fm	<p>This is optimized for printing in 8.5x11 format from a .PDF. Use it for books that are not commercially printed.</p> <ul style="list-style-type: none"> ◆ This is the default template for all reference manuals, WPS inline help, and Application Gateway. ◆ Re-apply this template to the Getting Started Guides or other manuals after preparing the file for commercial printing. See Converting to 8.5x11 Format on page 162. 	\\Admin\DocProcedures&Templates\Current Master Templates
5x8 Template.fm	<p>This is optimized for printing in 5.5x8.5 format. Use it for Getting Started Guides that are commercially printed. Apply it only when preparing a file for printing. See Converting to 5x8 Format on page 160.</p>	\\Admin\DocProcedures&Templates\Current Master Templates
7x9 Template.fm	<p>This is optimized for printing at 82% and trimming to 7x9 format. Use it for books (other than Getting Started Guides) that are commercially printed. Currently, this includes WfWI and Wise Installation System references only.</p> <p>Apply this template only when preparing a file for printing. See Converting to 7x9 Format on page 161.</p>	\\Admin\DocProcedures&Templates\Current Master Templates
A4 Small Template.fm A4 Large Template.fm	<p>These are not currently used, but should be kept up to date. They correspond to the 7x9 and 8.5x11 templates. See A4 Page Size on page 83.</p>	\\Admin\DocProcedures&Templates\Current Master Templates

Active Help Templates

For a current list, see [WebWorks Templates](#) on page 188. If you add a new style to the master FrameMaker template, be sure to add a corresponding style and mapping to the help templates.

Obsolete Templates

Template Name	Description	Location in Technical Publications
Robohelp.dot	The master Word template for WinHelp 4 help projects, which are now obsolete. Each help system's RoboHelp.dot template is stored in its Help project folder. If you open a WinHelp 4 project, first copy the project's robohelp.dot to your Microsoft Office\Templates\ folder.	<p>\Admin\OLD Help Templates\OLD RoboHelp.dot WinHelp tmp</p> <p>Instructions for WinHelp 4 projects are in \Admin\DocProcedures&Templates\OLD WinHelp Procedures</p>
Frame-to-Help Template.fm	The pre-conversion template for Frame files that are being converted to WinHelp 4 help.	<p>\Admin\OLD Help Templates\OLD Frame-to-Help WinHelp tmp</p> <p>Instructions for WinHelp 4 projects are in \Admin\DocProcedures&Templates\OLD WinHelp Procedures</p>
Frame-to-HTML Template.fm	A special FrameMaker template to globally reformat the Frame files before their conversion to HTML Help. It sets the formatting to the same styles that will be used in the help project, and is the basis for the style sheets that RoboHelp will create for the help project.	\Admin\OLD Help Templates\OLD Robo HTML templates + files
Wise.css	A cascading style sheet that you apply to the HTML files that are created from the Frame files. This fixes formats that do not convert from Frame, particularly the rules above and below notes.	\Admin\OLD Help Templates\OLD Robo HTML templates + files
Wise.htt	A RoboHelp template that you apply to the HTML files to add a header and footer to each topic. Wise.htt automatically applies the style sheet Wise.css. If you rename Wise.css you must change the reference in the template.	\Admin\OLD Help Templates\OLD Robo HTML templates + files

Differences Between Frame Templates

The following table lists the main differences between our Frame templates. Style definitions are different for each template but are not listed here.

Tip

If you're working in a Frame document and you're not sure what format it has, a quick way to find out is to check the font size of the B Body text.

Format	8.5x11 Template	5x8 Template	7x9 Template
B Body font size	8.5 pt	8 pt.	10 pt. (reduces to 8.2 pt.)
Page size in Frame	8.5x11	5.5x8.5	8.5x11
Output page size	8.5x11	5.5x8.5	7x9 (print at 82%)
Maximum column width (affects anchored frames and tables)	6.9	4.3	6.9
Text frames on master pages offset left and right for binding?	no	yes	yes
pagination	Single Sided, Delete Empty Pages	Double Sided, Make Page Count Even	Double Sided, Make Page Count Even

Changing the Frame Templates

Even the simplest change can affect many files, so do not change any of the Wise templates unless you discuss the change with the Technical Publications Manager. The template for training materials is based on the documentation template, so discuss any changes with the Curriculum Developer also.

In general, make all initial template changes in the master template files. From there, import the formats from the templates to the appropriate documents. For a general procedure for converting Frame documents to a new template, see [Guidelines for Frame Template Conversions](#) on page 78.

Where to Apply Template Changes

Every time you change the documentation template, you must apply the change to:

- ◆ All master templates. See [About Frame and Help Templates](#) on page 74.
- ◆ DocProcedures.book.
- ◆ All current Wise documentation. Discuss with the Technical Publications Manager how and when these should be updated. We might decide to wait until the next release of each product.
- ◆ Condition templates for Wise documentation.
- ◆ Any other documentation based on the master templates.

For a checklist of Frame documents that typically need to be updated, see the file List of Books.xls, located in Technical Publications\Admin\Notes Memos + Misc.

Updating the Help Templates

Some template changes also affect the help templates. (Examples: adding or deleting style tags, changing font sizes.) Typically, minor line spacing adjustments do not affect the help templates. See [Creating and Editing WebWorks Templates](#) on page 189

Applying a Format Change to a FrameMaker Book

The easiest way to apply a format change to multiple files is to use Frame's Import > Formats feature. However, there are a few instances in which you must be careful about what you import.

- ◆ Do not import page layouts between the IX file and other chapters. The index page layout is different from the regular chapter files.
- ◆ Importing new formats does not delete old ones.
Example: Suppose the old template contained a paragraph style named Oldstyle. You delete that style and replace it with one called Newstyle. When you import the formats to other files, those files will still contain the Oldstyle format. To remove the Oldstyle format, make sure it isn't used anywhere and then delete it from every file. Frame plug-ins can facilitate this task. See [Deleting Unwanted Style Tags](#).

Deleting Unwanted Style Tags

When you update the master templates, delete unused style tags to keep the style catalogs uncluttered. We use Frame plug-ins to accomplish this. The plug-ins are in Technical Publications\Admin\Documentation Software\Framemaker Plug-ins. For a list of plug-ins and how we use them, see [Tools for Creating Wise Documentation](#) on page 69.

5x8 Format Considerations

The 5x8 template has a maximum column width of 4.3 inches, which is much smaller than in the other book formats. Therefore, when you are working in a book that will be delivered in 5x8 format, the maximum width for anchored frames and tables is 4.3 inches, even if the files are currently in a larger format.

If you use the Table 2 styles, the maximum width is 4.05 because the table is indented .25 inch. Do not use the Table 3 styles.

The 5x8 format is only used for printed Getting Started Guides.

Also see [Converting to 5x8 Format](#) on page 160.

Guidelines for Frame Template Conversions

This section provides general steps for converting Frame documents to a new template. Depending on the complexity of your conversion, you might need to eliminate some steps or add steps.

Before You Start

1. Install these FrameMaker plug-ins. For sources, see [Tools for Creating Wise Documentation](#) on page 69.
 - Toolbox Basic Version by Systec
 - CudSpan Doc Tools - Hunt Overrides
 - Character Tools by Silicon Prairie Software
 - Paragraph Tools by Silicon Prairie Software
2. Obtain the following files:
 - Old template files
 - New template files
 - Book files to convert
3. Create conversion documents. See [Creating Conversion Documents](#).
 - Paragraph conversion document
 - Paragraph tag list
 - Character conversion document
 - Character tag list

See:

[Converting Books to a New Template](#)
[Checking the Document After a Template Conversion](#)

Creating Conversion Documents

To help you convert Frame documents to a new template, use the Silicon Prairie tools to create files that map new styles to old ones and list styles to keep.

1. Open the old template in Frame.
2. Select Tools menu > Paragraphs > Create Conversion Document.
3. On the FrameMaker dialog that appears, select **All Paragraph Tags** and click OK.
The tool reads the template and creates a new Frame file listing all paragraph formats in a 2-column table.
4. Switch to the new Frame document. In the Destination Tag column, enter the new paragraph tag that corresponds to the old one in the Source Tag column. Be sure to enter a new tag for every old tag listed.
5. Save the document.
6. Open the new template in Frame.
7. Select Tools menu > Paragraphs > Create Paragraph Tag List.
8. On the FrameMaker dialog that appears, select **All Paragraph Tags** and click OK.
The tool reads the template and creates a new Frame file containing all paragraph formats in a table. These are the paragraph tags you want to keep.
9. Save the document.

10. If you will convert character styles, repeat this procedure using the Characters tool.

Also see:

[Guidelines for Frame Template Conversions](#)
[Converting Books to a New Template](#)
[Checking the Document After a Template Conversion](#)

Converting Books to a New Template

Follow this procedure for each book to be converted.

Caution

This procedure contains several book searches. Because Frame often crashes after a book search, save the book frequently.

1. Make a backup of the book you are converting.
2. Open the book in Frame and follow each of the following procedures.

Eliminate Overrides

1. Select CudSpan Doc Tools menu > Hunt Pgf Overrides for Document.
The tool creates a new Frame document that lists all paragraph overrides.
2. Click an override in the list to go to the paragraph containing the override.
3. Apply the appropriate style to the paragraph to eliminate the override.
4. Repeat for the remaining overrides in the list.
5. Close the override list without saving.
6. Rerun the Hunt Pgf Overrides to verify that you fixed all the overrides.
7. If you think the book contains character style overrides, do one of the following:
 - Select Tools menu > Characters > Remove Overrides. This removes all character overrides in the book without showing them to you first.
 - Open each file in the book and select Tools menu > Characters > Find Next Override. This finds each override and lets you decide how to fix it.

Import New Formats

1. Open the new template file in Frame.
2. Make the book file active and select all files in the book.

Note

If the index, TOC, or other file in the book uses a different page layout, you might have to apply formats to that file from a different template in a separate step.

3. Select File menu > Import > Formats.
4. On the Import Formats dialog, select the new template file, mark the formats to import, and click Import.
5. If the new template contains different page layouts, you might need to apply new master pages.
6. Save the book.

Update Cross-References, Tables, and User-Defined Variables

During this procedure, deleted formats are written to the Frame Console.

1. If you are changing cross-reference formats:
 - Search the book for any cross-reference.
 - When a cross-reference is found, apply a new format as appropriate.
 - When you have converted all cross-references, select Toolbox menu > Formats > Cross-Reference Formats > Delete unused ones in catalog.
 - Save the book.
2. If you are changing table formats:
 - Search the book for any table.
 - When a table is found, apply a new table style as appropriate.
 - If page margins have changed, you might need to resize the table.
 - When you have converted all tables, select Toolbox menu > Formats > Table Formats > Delete unused ones in catalog.
 - Save the book.
3. If you are eliminating or changing user-defined variables:
 - Search the book for any variable.
 - When a variable is found, convert it to text or insert a different variable as appropriate.
 - When you have fixed all variables, select Toolbox menu > Formats > User Defined Variable Formats > Delete unused ones in catalog.
 - Save the book.
4. In the preceding steps, you might have deleted valid formats, if they didn't happen to be used in this book. To ensure that all the new formats are in the book file, whether or not they are used, re-import formats:
 - Select all files in the book and select File menu > Import > Formats.
 - On the Import Formats dialog, select the new template file, mark the formats to import, and click Import.

Note

If the index, TOC, or other file in the book uses a different page layout, you might have to apply formats to that file from a different template in a separate step.

5. Save the book.

Convert Old Styles to New Styles

1. Open the paragraph conversion document that you created earlier.
2. Make the book file active and select all files in the book.
3. Select Tools > Paragraphs > Convert Paragraphs.
4. Open the character conversion document that you created earlier.
5. Make the book file active and select all files in the book.
6. Select Tools > Characters > Convert Characters.
7. Close the conversion documents.
8. Open the paragraph tag list that you created earlier.
9. Make the book file active and select all files in the book.

10. Select Tools > Paragraphs > Remove Non-list Paragraph Tags.
11. Open the character tag list that you created earlier.
12. Make the book file active and select all files in the book.
13. Select Tools > Characters > Remove Non-list Character Tags.
14. Close the tag list documents.

Regenerate the TOC

When you rebuild the TOC in this step, the TOC entries are created using the new styles.

1. In the book file, right-click the TOC file and select Set Up Table of Contents.
2. On the dialog that appears, remove all entries from the Include Paragraphs Tagged list and click Set.
3. On the Update Book dialog, click Cancel.
4. Right-click the TOC file again and select Set Up Table of Contents.
5. Move the appropriate tags to the Include Paragraphs Tagged list and click Set.
6. On the Update Book dialog, click Update.

Also see:

[Guidelines for Frame Template Conversions](#)
[Creating Conversion Documents](#)
[Checking the Document After a Template Conversion](#)

Checking the Document After a Template Conversion

What you need to check will depend on what you are converting. Here are some suggestions:

- ◆ Page through the document. Look for:
 - Tables and anchored frames that need to be resized.
 - Incorrect font sizes. This is often caused by a character style or override.
 - Incorrect page layout.
 - Text that didn't get formatted properly because of a conversion problem or undiscovered override.
 - Text that needs to be placed on a new line or that needs to be merged with another line. Example: In some document layouts, the word "Note" is placed on a line above the note text; in other document layouts, it is run in with the note text. Depending on your particular layout, you might need to change the note heading one way or the other.
- ◆ Check for overrides.
- ◆ Verify that the book contains all the new formats but doesn't contain any old ones. You can:
 - Visually review the Paragraph and Character catalogs.
 - Use Silicon Prairie tools to list character and paragraph usage in the book or generate tag lists.
- ◆ Generate the TOC and index and verify that they are formatted correctly.

Also see:

[Guidelines for Frame Template Conversions](#)
[Creating Conversion Documents](#)

Converting Books to a New Template

A4 Page Size

The standard US Letter paper size (8.5 x 11) is not standard elsewhere. Outside the U.S., the A4 paper size (8.268 x 11.693) is more common. Most articles that discuss preparing documentation for localization recommend using the A4 page size, or at least using an alternate A4 template to convert existing documents at translation time.

Wise documents convert easily to the A4 size; we simply change the page size and make some adjustments on the master pages. We are able to keep the text frames the same size as in the standard template, which means that the text will flow exactly the same regardless of which template is used.

Two A4 templates are in the Technical Publications directory under Admin\DocProcedures&Templates\Current Master Templates. Except for the page settings, they are the same as the corresponding master templates. If you change the master template significantly, you might want to create a new A4 template; see [Creating an A4 Template](#) on page 84.

How the A4 Template Works

Following are the results of tests made with an A4 version of our doc template.

◆ **Printing from Frame**

When you print an A4 document from Frame at 100%, click OK at the prompt that tells you that "pages are larger than the printer paper".

◆ **Making a .PDF file**

Make the .PDF as usual, leaving the page size at 8.628 x 11.693. The resulting .PDF will be in A4 format. You can easily print the file on 8.5 x 11 paper without checking any additional options.

◆ **Making a PostScript file for 7 x 9 printing**

This does not work. The US Letter page size easily reduces to 7 x 9. However, the A4 page size has different proportions so there is no single reduction percentage that will result in a 7 x 9 page. We need to find out if there is another standard for computer manuals that is proportionate to the A4 page size.

When to Use the A4 Template

We could use the A4 page size as our standard, except for not being able to convert to 7 x 9 format. That's a significant problem, therefore the US Letter size will continue to be our standard. In the future, when we need to localize our documentation, we can apply an A4 template to the files we deliver to the translator.

Occasionally, our Netherlands office has asked for a .PDF they can use to print manuals in Europe. They were able to use the US Letter-formatted files we sent them, but in the future we should send them A4-formatted files. Otherwise, we currently do not need to use the A4 templates.

How to apply the A4 template:

1. In Frame, open the A4 template and open a book.
2. Make the book active and select all files in the book.
3. Select File > Import > Formats.
4. In the Import Formats dialog, select the A4 template file in the **Import from Document** drop-down list.
5. Mark the **Page Layouts** checkbox and clear all other options.
6. Click the Import button.

7. Save and close the book.

Creating an A4 Template

This section lists the settings to use to convert our standard template to the A4 paper size (8.268" x 11.693"). The size of the text frames remains the same so that the text flows exactly the same as when using the standard template.

To create an A4 template:

1. Make a copy of the current Master Template and open the copy in Frame.
2. Select View menu > Master Pages.
3. Select Format menu > Page Layout > Page Size.
The Page Size dialog appears.
4. From **Page Size**, select **A4 Letter**.
5. Change the Offset From settings for each master page:
 - On the first master page, click in the first frame on the page (the header frame) and select Format menu > Customize Layout > Customize Text Frame. (Keyboard shortcut is Alt+o, u, c.)
 - On the Customize Text Frame dialog, change the **Offset From** settings (see the table below) and click Set.
 - Repeat for the remaining frames on the page.
 - Move to the next master page and repeat the process.
6. Save the A4 template.

Offset Settings for A4 Master Pages

Each master page contains 3 text frames: the header is the narrow frame at the top of the page; the main body is the large frame that takes up most of the page; and the footer is the narrow frame at the bottom of the page.

Master Page	Frame	Offset From Top	Offset From Left
Left	header	.946	.484
	main body	1.546	.484
	footer	10.771	.484
Right	header	.946	.884
	main body	1.546	.884
	footer	10.771	.884
Chapter Title	header	.946	.884
	main body	1.546	.884
	footer	10.771	.884

The Chapter Title master page is the same as the Right master page, except that the header content is different.

Also see [A4 Page Size](#) on page 83.

Chapter 5

Documentation Structure and Formatting

This section covers the following topics:

- ◆ [Formatting Text in FrameMaker.](#)
- ◆ [Formatting Headings.](#)
- ◆ [Using Character Tags.](#)
- ◆ [Using Change Bars.](#)
- ◆ [Formatting Lists.](#)
- ◆ [Using and Formatting Notes.](#)
- ◆ [Anchoring Tables.](#)
- ◆ [Forcing Line Breaks.](#)
- ◆ [Creating Screen Shots and Placing Graphics.](#)
- ◆ [Hyperlinks.](#)
- ◆ [Using Conditional Text.](#)
- ◆ [Inserting Temporary Author's Notes.](#)
- ◆ [Multilingual Production Requirements.](#)

Formatting Text in FrameMaker

The current Frame template for Wise documentation contains standard formatting tags (styles) for paragraphs, characters, tables, cross-references, and most other situations. Use these standard tags to format *all* text. Do not override the standard formats.

Do not create new paragraph, character, table, cross-ref, or any other tags unless you discuss the change with the Technical Publications Manager. There might already be a standard way of dealing with the situation for which you are contemplating a new paragraph tag. See [Changing the Frame Templates](#) on page 76.

Using Quick Tagging

Tags in our Frame template are named to facilitate quick tagging from the keyboard.

- ◆ To quick-tag a paragraph, press F9, type the first few letters of the tag name, and press Enter.
- ◆ To quick-tag a selected character, press F8, type the first few letters of the tag name, and press Enter.

Formatting Headings

This section discusses the use of headings and illustrates the heading styles.

See:

[Proper Use of Headings](#)
[Using H1s That Look Like H2s](#)
[Forcing a Line Break in Headings](#)

Proper Use of Headings

Do not place headings back to back with no body or other text in between. This might involve reworking material in chapters. Example:

(ChapterTitle) Tools

(H1) About the Tools

(H2) SetupCapture

(B Body) The SetupCapture tool and so on.

To fix the preceding example, you probably would add introductory material after ChapterTitle and after H1 to avoid having "textless" headings, that is, headings followed by more headings instead of being followed by body text. You do this because each ChapterTitle, H1 Heading1, H2 Heading2, and H3 Heading3 in the book becomes a *separate topic* in online help. Therefore, a textless heading in the book becomes a textless topic in help.

Also see [Topic Heading Guidelines](#) on page 13.

Using H1s That Look Like H2s

As of December 2003, these are not used. We fixed the help template to eliminate the problem in the Help TOC. - JK

In some chapters, specifically those that document script actions, we need to go from the ChapterTitle to an H2 instead of an H1. We do this because the topics are numerous but brief, and we don't want to break a page at every topic. However, this causes an empty TOC book in the help. To avoid this, we use the style **XH1 Heading1**. It's an H1 style that looks and behaves like an H2. The Frame and help conversion templates treat this style like an H1 and create the proper TOC level. When you create the book .PDF, you must remember to include this style in the Bookmark list, and give it the same Bookmark Level as the **H1 Heading1** tag.

When an XH1 topic contain subtopics, use the style **XH2 Heading2** for the subtopics.

Forcing a Line Break in Headings

When a heading is so long that it wraps to a second line, you might be tempted to force a line break at the most meaningful and attractive point in the heading. DON'T BOTHER. Here's why:

- ◆ Using Shift+Enter to insert a forced return causes an unattractive line break in the book's table of contents. This means that, every time you build the TOC for a .PDF or export to help, you must remember to fix those line breaks manually.

- ◆ Using nonbreaking spaces in the heading instead of a forced return controls the line break nicely, but the nonbreaking spaces appear as strange characters in the .PDF Table of Contents (in Acrobat 5).
- ◆ Changing the right margins of the heading tag for a specific heading can break the line where you want, but we do not allow format overrides.
- ◆ Because a book's format can change depending on the deliverable, a line break that looks good in 8.5x11 format might not work in 5.5x8.5 format.

Exception:

Sometimes, a heading is so long that it flows onto two lines in the TOC, usually unattractively. In the past, we would fix these by inserting forced returns in the TOC after each build. Instead, you CAN use Shift+Enter to force spaces in the heading itself, which will control where the line breaks in the TOC.

Using Character Tags

<<<Much of this information has been moved to the new template file, which will be implemented in Q1 2005. JK>>>

The Wise documentation template uses these character tags: Blue, Emphasis, Field, Keycap, Script, and several Symbol tags.

In February 2003, we redesigned the character tags to be more generic so that they can be used within any paragraph style. Previously, the character tags specified the font size, so we needed extra character tags to apply to small text. (Example: We used the EmphasisSmall tag to add italics to notebody text. Now, all character tags will work with any paragraph style, regardless of type size.) The tags that were removed from the template are: Blue, BlueCallout, BlueSmall, EmphasisSmall, and NoteField.

Script

Note

Per Abreve standards, we have discontinued the use of the Script tags. They are still in the templates but we will probably remove them eventually. 1/2004.

For script lines that appear in the product, use the paragraph tags S1 Script1, S2 Script2, or S3 Script3, depending on where your indent level is. S1 Script1 fits underneath body text.

```
Include Script C:\Program Files\OpensWord.wse
```

- ◆ Use the tag S2 Script2 if you need script underneath a numbered list (or bulleted list using B1 Bull1):

```
Include Script C:\Program Files\OpensWord.wse
```

- When you need script underneath a level 2 bullet (tag B2 Bull2), use the tag S3 Script3:

```
Include Script C:\Program Files\OpensWord.wse
```

Use the S4 Script4 tag when the script lines are very long as shown below.

```
' assume tblFeature is the Feature WTable
Dim row As WRow
For each row in tblFeature.WRows
' set the description field to the feature name + "Description"
row("Description") = row("Feature") & " Description"
' OR
' row("Description") = row(0) & " Description"
Next
```

If you need to number script lines, in any format, do so manually. When you number script lines, include the statement "lines are numbered for clarity" in the text

Sometimes you might want to include a section of script within a paragraph of body text. In that case, use the character format named Script. Example: Enter the filename OpensWord.wse at the prompt.

Wide Tags

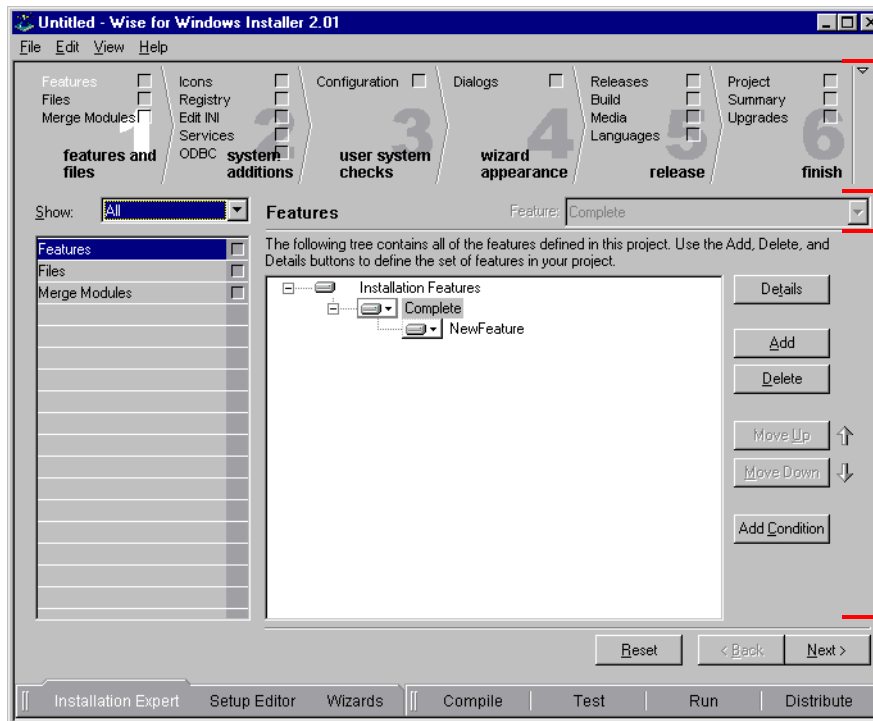
In general, we want to maintain the wide left margin in our books. Occasionally, however, you need to use the entire page width.

Use the WideBody tag for text that has no left indent. This is rarely used in body text. It is mainly used in graphic callouts when you need a larger font than the Callout tag provides. This tag has the same line spacing as B Body.

To place a graphic that spans the entire page width, use the WideGraphic tag. Its line spacing is different from the WideBody and is more appropriate for use with anchored frames.

The following example uses the WideGraphic paragraph tag for the anchored frame and the Callout tag for text pointing to a graphic.

In this example, the WideBody tag provides larger type than the usual Callout tag.



(This is Callout text.) Step buttons. Each step is made up of several pages. The first few page names appear on the button.

Here's an example of using bullets in callouts:

- ◆ CalloutBull1
 - CalloutBull2.

Using Change Bars

We use change bars to identify new or changed text in Frame documents. This helps reduce technical review time by letting the reviewers concentrate only on the material that has changed.

When to Use Change Bars

Typically, you turn on change bars for the entire book as soon as you begin development for a new release. Keep them turned on so all changes for the release are recorded.

When Not to Use Change Bars

If you are making format or style changes in a book that do not affect the book's content, and if you do not have to make the same changes in another, similar book, you do not have to use change bars. Example: When we decided not to use the word *button* with the button name, we turned off change bars before making those changes, because technical reviewers did not need to see them.

Also see:

[Turning Change Bars On and Off by Book or Chapter](#)
[Clearing Change Bars in an Entire Document or Book](#)
[Removing and Applying Change Bars in Selected Text](#)
[Searching for Change Bars](#)

Turning Change Bars On and Off by Book or Chapter

1. Open the appropriate document or book. If you open a book, select the documents to affect.
2. Select Format > Document > Change Bars.
3. In the Change Bar Properties dialog that appears:
 - To turn change bars ON, mark **Automatic Change Bars**.
 - To turn change bars OFF, clear **Automatic Change Bars**.
4. Leave the defaults in the rest of the dialog and click Set.

When you make a change in a document with change bars turned on, a change bar is applied.

Also see [Using Change Bars](#).

Clearing Change Bars in an Entire Document or Book

Caution

Make sure you really are ready to clear change bars because you cannot undo this operation.

1. Open the document or book. If you open a book, select the documents to affect.
2. Select Format > Document > Change Bars.
The Change Bar Properties dialog appears.
3. Mark **Clear All Change Bars (No Undo)**.
4. To turn off change bars at the same time, clear **Automatic Change Bars**.
5. Leave the defaults in the rest of the dialog and click Set.

All change bars are removed from the selected documents.

Also see [Using Change Bars](#).

Removing and Applying Change Bars in Selected Text

Sometimes you need to make a small change, such as deleting an extra space or changing a comma to a semicolon. If change bars are turned on, a change bar appears. Because this could be confusing to the technical reviewers, you might want to remove the individual change bar.

In rare instances, you might need to apply a change bar to selected text. Example: You change a paragraph in a book before realizing that you forgot to turn on automatic change bars.

To remove change bars from selected text:

This method might remove character formatting. If the change bar text also has character formatting, see the procedure below.

1. Select the text to remove the change bar from. Select more than the original change. (Example: If you changed a few words, select the entire sentence; if you changed a sentence or more, select the entire paragraph.) This ensures that you include all the changed text.
2. Press F8 + Enter. This applies the default paragraph font.

To remove change bars without removing character formatting:

1. Select the text to remove the change bar from. Select more than the original change. (Example: If you changed a few words, select the entire sentence; if you changed a sentence or more, select the entire paragraph.) This ensures that you include all the changed text.
2. Select Format > Style > Change Bar.
A change bar is applied to the entire selection.
3. With the text still selected, select Format > Style > Change Bar again.
The change bar is removed.

Why do the extra step of applying a change bar to additional text? Because unless you select *exactly* the same text to which the change bar was applied originally, the change bar is not removed. Then you have to do the above apply/remove steps anyway.

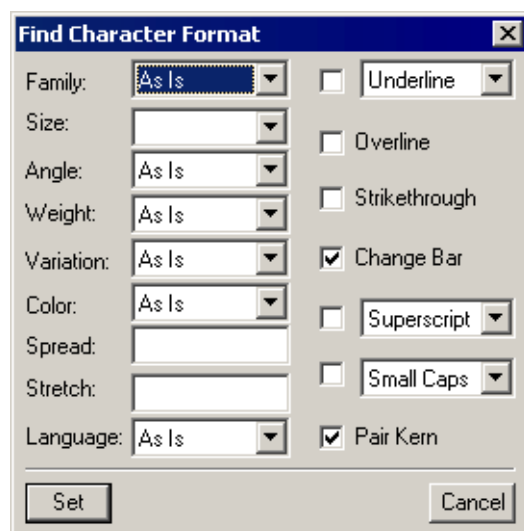
To apply change bars to selected text:

1. Select the text to add the change bar to.
2. Select Format > Style > Change Bar.
A change bar is applied to the selection.

Also see [Using Change Bars](#).

Searching for Change Bars

1. Open the Find/Change dialog in FrameMaker.
2. From **Find**, select **Character Format...**
The Find Character Format dialog appears.
3. Complete the dialog as shown in the following screen shot.



4. Click Set.
5. In the Find/Change dialog, click Find.

All change bars except for those applied to cross-references are found. Generally, the ones applied to cross-references are not relevant anyway.

[Also see Using Change Bars.](#)

Formatting Lists

<<<Move some of this to the new template file, which will be implemented in Q1 2005.
JK>>>

The FrameMaker template contains paragraph tags for a numbered list and 3 levels of bulleted lists.

Numbered Lists

We typically use numbered lists for procedures and processes. Most other lists are bulleted.

1. Use the tag Numbered for a numbered list.

When you need to add a second paragraph to a numbered item, use the tag ON BodyInNumber. This maintains the proper indent and leaves a little extra space between the paragraphs.

2. Then you can go back to the numbered list style.

Bulleted Lists

There are 3 levels of bullets that can be used together or with other paragraph tags.

- ◆ B1 Bull1. Use this for bullets following body text or standing on their own (not subordinate to other bullets or numbered lists.

To add a second paragraph to a bulleted item (B1 Bull1), use the tag O1 BodyInBull1.

- ◆ **B1B Bull1 Bold**

Use this to format controls in a dialog. See [Typographic Conventions](#) on page 25.

To use a bulleted list within a numbered list, use the tag B2 Bull2.

1. Here's a numbered item followed by a bulleted list.
 - B2 Bull2 can be used as a subordinate to either the Numbered or B1 Bull1 style. To add a second paragraph to a bulleted item (B2 Bull2), use the tag O2 BodyInBull2.
 - **B2B Bull2 Bold**
Use this to format controls in a dialog. See [Typographic Conventions](#) on page 25.
2. And back to a numbered item.

Use B3 Bull3 for a more indented bullet, or as a subordinate to the style Indented2 or even B2 Bull2.

Here's an example of using the B3 Bull3 style under indented text:

- Here's one bullet using the style B3 Bull3.
To add a second paragraph to a bulleted item (B3 Bull3), use the tag O3 BodyInBull3.
- **B3B Bull3 Bold**
Use this to format controls in a dialog. See [Typographic Conventions](#) on page 25.

Example: Nested Bullets

- ◆ B1 Bull1
 - B2 Bull2
 - B3 Bull3

Non-Bulleted Lists

In a non-bulleted list, use BT BodyTight to get a tighter list. Example:

ACT 3, 4
Adobe Acrobat Reader 3.01
Adobe Pagemaker 6.0, 6.5
Adobe PageMill 3.0
Adobe PageMaker 6.0
Adobe Photoshop 5.0
Attachmate Extra 6.4

For a tight indented list without bullets, use Indented2Tight. This tag has the same indent as a Bull2. Example:

ACT 3, 4
Adobe Acrobat Reader 3.01
Adobe Pagemaker 6.0, 6.5
Adobe PageMill 3.0
Adobe PageMaker 6.0

You sometimes have to insert an extra return after BT BodyTight and Indented2Tight text. When you do, also include a nonbreaking space (press Ctrl+Spacebar) so that the blank line will not be ignored in the conversion to help.

Using and Formatting Notes

Notes call the user's attention to information of special importance or information that can't otherwise be presented in the main text. Use notes sparingly so they remain effective attention-getters.

Per Abreve standards, we eliminated the graphic symbols that we used to use with note headings, except for the edition caveats.

We use several different types of notes, based loosely on recommendations in *Microsoft Manual of Style for Technical Publications*. They are: notes, cautions, and edition caveats.

Notes

A note indicates neutral or positive information that emphasizes or supplements important points of the main text. It supplies information that may apply only in special cases. Use notes to draw the users' attention to additional material. Example: When discussing WebDeploy's FTP function, use a note to tell users that Windows includes a built-in FTP client they can use for testing.

You can also use a note to provide tip information. A tip helps users apply techniques and procedures described in the text to their specific needs. Suggests alternative methods that might not be obvious and helps users understand the benefits and capabilities of the product. Prior to February 2003, we had a separate graphic for tips. Now, just use a note.

Three formats are provided for notes, to allow for different levels of indent. Do not use a colon after the note heading.

Cautions

A caution advises users that failure to take or avoid a specified action could result in loss of data.

Use the same formats for cautions as for notes. Do not use a colon after the note .

A warning advises users that failure to take or avoid a specific action could result in physical harm to the user or the hardware. Warnings are rarely if ever used in Wise documentation. In February 2003 we eliminated the use of warnings. Use a caution instead.

Edition Caveats

An edition caveat indicates that a feature is only available in certain editions of the product. Put an edition caveat in every single topic where it applies, because users do not typically start at the beginning of a chapter and then move forward. They might start with the index or the TOC and then turn directly to a section within a chapter. This is even more crucial when the book is converted to help. We have to assume that the reader sees each topic independently of all other sections. By "topic," we mean sections preceded by H1, H2, and H3 paragraph tags.

Edition caveats always appear at the beginning of the topic, just below the topic heading. If you need to point out a single element that is edition-specific, incorporate it into the text. Example:

In the Enterprise Edition, a third tab appears, which ...

We use 2 types of edition caveats. Use the style Edition, shown below, to state that a feature is available in certain editions only. Use a period at the end of the statement.

 *Professional Edition only.*

Use the style Edition2 when the edition caveat requires additional explanation or when the restriction is not necessarily edition-related. Examples:

 *Professional and Enterprise Editions only.*

Virtual Capture is available in the Enterprise and Professional Editions only. This technology is supported only by Windows NT 4, Windows 2000, and Windows XP.

 *IBM Tivoli only.*

This feature is available only if IBM Tivoli Configuration Manager version 4.1 or 4.2 is installed on the same computer that is running Wise Package Studio.

Anchoring Tables

When you add a table to a FrameMaker document, anchor the table to a new line. If you anchor it to a line that contains text, that text can end up in the wrong place when the Frame file is converted to help. It's tempting to anchor a table on a text line because it produces more attractive spacing, but it causes problems later. (This is not as critical in HTML help as it was in WinHelp. In HTML, a table is always forced onto its own line.)

Caution

When you add a table to a book that will be delivered in 5.5x8.5 format, do not make the table wider than 4.3 inches. That is the maximum column width in that format. If you use the Table 2 styles, the maximum width is 4.05 because the table is indented .25 inch. Do not use the Table 3 styles.

Also see <<<x-ref to table tags section in template>>>.

Forcing Line Breaks

Use forced line breaks sparingly. It is tempting to use them to fine-tune the book formatting but, when you convert the text to help, the results are rarely satisfactory. Typically, we use forced line breaks in the following instances:

- ◆ To break long heading lines, but only those that will wrap in the book's TOC. See [Forcing a Line Break in Headings](#).
- ◆ To start a new paragraph when you don't want the extra space that normally occurs between paragraphs. This typically is used in lists and to break paragraphs within table cells. [We might discontinue this usage - JK 5/03]

Do not force a line break in the middle of a paragraph. This often has been done to make a long URL or a sample script line wrap properly. The text ALWAYS wraps differently in help.

Creating Screen Shots and Placing Graphics

Guidelines

- ◆ Paint Shop Pro is the preferred tool for capturing screen shots, however, contractors can use the tool of their choice as long as they follow the screen color and resolution guidelines in this section.
- ◆ Make sure all anchored frames are on their own line, not anchored to a paragraph with text. Otherwise, if you try to hide the image from help with conditional text, you might end up hiding the entire paragraph too.
- ◆ Import all images by reference only. DO NOT use Copy/Paste to copy the image and paste into Frame. This will store the image in the document instead of storing it as a separate referenced file, bloating the file and preventing future modifications to the image.
- ◆ If an image should not appear in the help, apply the **not for help** conditional text tag; see [How to Use the not for help Tag](#) on page 110. The kinds of images we usually keep for help are:
 - Alert images (for notes, tips, and so on).
 - Inline images that illustrate icons or buttons.
 - Overview screen shots that illustrate a portion of the product interface, usually with callouts.
 - Any other images that help explain a topic.

In July 2002, we decided to take a minimalist approach to screen shots. For guidelines on when to use a screen shot, see [Use of Screen Shots](#) on page 27.

Also see:

[Setting Screen Colors for Screen Shots](#)
[Using PaintShop Pro for Screen Shots](#)
[Placing an Image in FrameMaker](#)
[Specifying the Format for Graphics in Help](#)
[Sizing Graphics for Help](#)

Setting Screen Colors for Screen Shots

Use the default Windows screen colors.

Sometimes it's necessary to edit screen shots in PaintShop. (Example: to remove the word "Beta" from the title bar.) To make it easier to edit the title bar, turn off the color gradation in your display settings. <<<Is this still valid for XP?>>>

1. From the Control Panel, open the Display Properties dialog.
2. Click the Appearance tab.
3. In the **Item** drop-down list, select **Active Title Bar**.
4. In the **Color** drop-down lists to the right of the item, make the **Color 2** the same as the first **Color**.
5. Repeat for the **Inactive Title Bar** item.

Also see [Creating Screen Shots and Placing Graphics](#).

Using PaintShop Pro for Screen Shots

To set up Paint Shop Pro for captures:

1. Select Capture > Setup.
2. In the Capture Setup Dialog,
 - Mark the **Window** option.
 - Mark the **Delay Timer** option and enter the number of seconds for the delay. Usually 2 seconds is sufficient.
 - Click OK.

To capture a screen with Paint Shop:

Capture screen shots at full screen size (typically 96 dpi).

1. In the Wise product, set up the screen or dialog.
2. Go to Paint Shop and press Shift+C.
3. Paint Shop switches to the last-active window, captures it, and places the captured image in the Paint Shop window.

Note

Keep your mouse pointer out of the way during the capture, unless it should be part of the image.

4. Save the image in .TIF format into the chapter's graphic folder.

If the chapter does not have a graphic folder, then create one. Example: If you are working in the Basics chapter and are adding a new graphic, store the graphic file in the Basics TIFs folder. This is to prevent the path from breaking later when transferring Frame files from computer to computer.

Also see [Creating Screen Shots and Placing Graphics](#).

Placing an Image in FrameMaker

Caution

When you add an image to a book that will be delivered in 5.5x8.5 format, do not make the anchored frame wider than 4.3 inches. That is the maximum column width in that format.

Make sure that your .TIF image is stored in the chapter's graphic folder before you place it in the Frame document.

1. Click in the document where you want to insert the frame.
 - When inserting a screen shot, click on a blank line. This is important when the document is converted to help.
 - When inserting an image of an icon, tool, or button, click in the line of text.
2. Select Special > Anchored Frame.
The Anchored Frame dialog appears.
3. Set options in the Anchored Frame dialog and click the New Frame button. For recommendations on frame settings for different kinds of images, see [Recommended Settings for Anchored Frames](#) on page 102.

4. Click the frame to select it, then select File > Import > File.
The Import dialog appears.
5. Navigate to the image file, verify that the **Import by Reference** option is marked, and click the Open button.
The Imported Graphic Scaling dialog appears.
6. Set scaling options:
 - For inline icons, tools, and buttons, use 96 dpi. If the image is too small to be readable, use 60 dpi to enlarge it.
 - For icons, tools, and buttons in a table, use 60 dpi.
 - When the screen shot must be readable, use 96 dpi. Example: A screen shot should be readable when it is provided to verify the screen state, mark the user’s progress in a procedure, or locate window elements and objects.
 - When the screen shot must be recognizable, use 140 dpi or less. This size is sufficient when user doesn’t have to be able to read the text. Example: When a screen shot shows an overview of the product interface.
7. Resize the anchored frame to fit snugly around the image.

Note

To resize (or “shrink-wrap”) the frame to the exact dimensions of the image, click the frame and press Esc+m+p.

8. If you resized the frame by dragging, open the Anchored Frame dialog and reset **Distance Above Baseline** to zero. If you skip this step, there will be extra (or not enough) spacing between the dialog and the text that follows it.
9. If the image is inline, you might need to <<<sentence never finished!!!>>>

Also see [Creating Screen Shots and Placing Graphics](#).

Recommended Settings for Anchored Frames

When you insert an anchored frame into a document, as described in [Placing an Image in FrameMaker](#) on page 101, you enter settings in the Anchored Frame dialog. The settings you enter depend on the kind of image you will import.

Caution

When you add an image to a book that will be delivered in 5.5x8.5 format, do not make the anchored frame wider than 4.3 inches. That is the maximum column width in that format.


Image on its Own Line

Frame provides a shortcut for these settings: select the anchored frame and press Esc+m+p. This “shrink wraps” the frame to the exact dimensions of the image

Frame Option	Value
Anchoring Position	At Insertion Point

Frame Option	Value
Distance above Baseline	0
Size	Enter estimated values and resize the frame after importing the image.

Inline Images Within a Text Line

Usually, this is done for icons and tools, like this  .

Frame Option	Value
Anchoring Position	Insertion Point
Distance above Baseline	Start with -4.0, but you'll need to experiment with this measurement after importing the image.
Size	Enter estimated values and resize the frame after importing the image.

Inline Images at the Beginning of a Text Line



In these cases, the image acts like a bullet. We rarely use this format anymore. Instead, we use a table, with the icon in the first column and its description in the second column.

Frame Option	Value
Anchoring Position	Run into Paragraph
Alignment	Left
Gap	0
Size	Enter estimated values and resize the frame after importing the image.

Note

Images that begin a paragraph look best in help if you build some spacing into the image itself instead of setting a value in the **Gap** field. Example: The .TIF file in the paragraph above has approximately 5 pixels of white space to the right of the actual image.

[Also see Creating Screen Shots and Placing Graphics.](#)

Specifying the Format for Graphics in Help

Our WebWorks templates are set up to automatically convert all graphics to .GIF format. This is the best format for most of our screen shots and other graphics. Occasionally, you might want a graphics to be converted to .JPG format. To do this, you need to add a GraphicStyle marker to the anchored frame to override the WebWorks default. The marker that causes the graphic to be converted to .JPG format also controls the size of

the converted graphics. (For instructions on sizing the default .GIF graphics, see [Sizing Graphics for Help](#) on page 105.)

When To Use Each Graphic Format

- ◆ .GIF format is best for images consisting of solid, flat colors, that is, colors without gradients. It is also best for images containing type. This includes all our screen shots with callouts.
- ◆ .JPG format is best for photos and images that have shaded or gradient colors. Few of our screen shots will contain gradients, so it will be rare for us to require this format.

To make a graphic convert to .JPG format:

Note

Do not try to add graphic markers to small icon graphics. It's too hard to add an additional frame to the tiny anchored frame. Besides, with images that small it's not necessary to make them .JPGs.

1. In your FrameMaker documents, create a custom marker type named GraphicStyle. (This is already included in the current template.)
2. Locate the anchored frame that contains the graphic you will make a .JPG.
3. Add a **GraphicStyle** marker to a text frame within the anchored frame:
 - If the anchored frame contains one or more text frames for callouts, add the marker to the first callout frame.
 - If the anchored frame does not contain callouts, insert a text frame within the anchored frame and, within the text frame, insert a GraphicStyle marker.
4. In the **Marker Text** field, type the name of the graphic style WebWorks Publisher should use when converting the graphic. The predefined graphic styles are as follows:

Marker Text	Produces a .JPG file at
JPEG	100% of original size
JPEG90	90%
JPEG80	80%
JPEG70	70%
JPEG60	60%
JPEG110	110%
JPEG120	120%

5. Add a comment above the anchored frame to indicate that it contains a GraphicStyle marker, and apply the **comment** conditional text tag to the comment text.
6. Save the FrameMaker file.
7. Regenerate the help.

Also see [Creating Screen Shots and Placing Graphics](#).

Sizing Graphics for Help

When WebWorks automatically converts graphics from .TIF to .GIF format, the resulting graphics are often too large when viewed in the help window. To adjust the size of graphics in help, you add a marker to a text frame within the graphic's anchored frame.

Note

Do not try to add markers to small icon graphics. It's too hard to add an additional frame to the tiny anchored frame.

There are two ways to size graphics for help, depending on the format of the converted graphic.

- ◆ .GIF graphics. Graphics converted by WebWorks default to .GIF format; most of the graphics in our documents will look best in this format. To resize .GIF graphics you use the `GraphicScale` marker and specify a percentage. See the steps below.
- ◆ .JPG graphics. In order to get a graphic converted to .JPG format in the first place, you must use the `GraphicStyle` marker. The JPEG style markers we have defined in our WebWorks templates let you specify any of several predefined percentages. See [Specifying the Format for Graphics in Help](#) on page 103.

It's best to wait until you have created the first help build before adding size markers to graphics. After you have seen how the graphics look in the help you can better determine whether they need to be resized.

To size a .GIF graphic by percentage:

1. In your FrameMaker documents, create a custom marker type named `GraphicScale`. (This is already included in the current template.)
2. Locate the anchored frame that contains the graphic to size.
3. Add a **GraphicScale** marker to a text frame within the anchored frame:
 - If the anchored frame contains one or more text frames for callouts, add the marker to the first callout frame.
 - If the anchored frame does not contain callouts, add a text frame. Insert a text frame within the anchored frame and, within the text frame, insert a `GraphicStyle` marker.
4. In the **Marker Text** field, type a number to represent the percentage to use for resizing. Do not type the percent sign. A percentage of 80 seems to work best for most of screen shots and other graphics.
5. Add a comment above the anchored frame to indicate that it contains a `GraphicScale` marker, and apply the **comment** conditional text tag to the comment text.
6. Save the FrameMaker file.
7. Regenerate the help.

Also see [Creating Screen Shots and Placing Graphics](#).

Hyperlinks

Use FrameMaker's Hypertext feature to create hyperlinks in Frame books. These links will be converted to hyperlinks in the help, and some of them will also work in the .PDFs. Use this feature to embed the following kinds of links:

- ◆ Links to URLs (examples: Wise's Web site or the MSDN library). See [Creating Links to URLs](#).
- ◆ Links to topics in the Windows Installer SDK Help. See [Linking to an SDK Help Topic in Wise Products](#) on page 107 and [Linking to a Topic in Visual Studio .NET](#) on page 108.

You also can link to topics in other Wise help files. To do this, create a cross-reference to the topic in the other book. See [Cross-References to Another Wise Book](#) on page 29.

Creating Links to URLs

Note

Follow these instructions exactly. Sometimes a hyperlink will look correct in the Frame file but in the help, only the first character of the hyperlink text is formatted as a hyperlink. The only thing we can think of is that the steps for applying the character tag and adding the marker were not followed precisely. If this happens, it is best to remove the character tag, remove the marker, and start over.

1. In Frame, type the URL in the text but do not apply any special formatting yet.
Example:

Search for "Interoperating with Unmanaged Code" in the MSDN Library
(msdn.microsoft.com/library/).
2. Select the URL and copy it.
3. With the URL text still selected, apply the character style Weblink. This displays the URL text in blue.
4. With the URL text still selected, select Special > Hypertext.
5. In the Hypertext dialog, select **Go to URL** in the **Command** drop-down list.
6. In the **Syntax** field, paste the URL after the text *message URL*, leaving a space after "URL".
 - Type `http://` after the text *message URL*, leaving a space after "URL".
 - Paste the URL after the final `/`.
 When you finish, the contents of the **Syntax** field should look like this:

message URL `http://msdn.microsoft.com/library/`
7. Click the New Hypertext Marker button.
 Your text will now look like this (if you have Text Symbols turned on, a marker will appear to the left of the URL text):

Search for "Interoperating with Unmanaged Code" in the MSDN Library
(msdn.microsoft.com/library/).

When you create the .PDF, and when help is created, the marker is converted to a hyperlink.

<<<There's another way to do this. If you use the character style WebJump, you don't have to add the marker; WebWorks creates the hyperlink automatically. The restriction

is that the URL must be in either `www.` or `http://` format. The downside is that the URL links won't work in the PDF. We need to decide which we want to use. JK 5/6/03>>>

Creating Links to External Help Topics

You can create links to topics in an external help system (example: the Windows Installer SDK Help or a Wise help system). For these links to work in the help, all referenced help files must exist in the same directory as the main help.

These links do not work in the .PDF. If an end user hovers the mouse over one of these links, the finger icon will appear, but the link will not work. Because the links are not colored blue in the .PDF, we have to hope the user won't think to try the links.

The procedure is the same regardless of the product you're documenting. However, when you link to a topic in Visual Studio .NET, you find the help topic path differently. See:

[Linking to an SDK Help Topic in Wise Products](#)

[Linking to a Topic in Visual Studio .NET](#) on page 108

Also see [Cross-References to Another Wise Book](#) on page 29.

Linking to an SDK Help Topic in Wise Products

 *This procedure applies to Windows Installer 3.0.*

Several Wise products contain references to SDK Help topics contained in MSI.chm. For guidelines on the form to use for referencing an SDK Help topic, see [Cross-References to a Third-Party Help System](#) on page 29.

Note

Follow these instructions exactly. Sometimes a hyperlink will look correct in the Frame file but in the help, only the first character of the hyperlink text is formatted as a hyperlink. The only thing we can think of is that the steps for applying the character tag and adding the marker were not followed precisely. If this happens, it is best to remove the character tag, remove the marker, and start over.

1. In Frame, type the reference in the text but do not apply any special formatting yet.
Example:
 See ScreenX Property in the Windows Installer SDK Help.
2. Open the SDK Help and display the topic.
3. Click anywhere in the SDK topic.
4. Select View > Source. The file opens in a Notepad window.
5. In Notepad, select File > Save As. The Save As dialog appears, with the file name in the **File Name** field.
6. Copy the file name to the clipboard, omitting the bracketed number at the end of the file name.

Example: If the **File Name** field contains this:

```
C:\Documents and Settings\joannk\Local Settings\Temporary Internet
Files\Content.IE5\WLIFCTIN\screenx_property[1]
```

you would copy this:

```
screenx_property
```

7. In Frame, select the topic title in the text. Example: if the text says:
 See ScreenX Property in the Windows Installer SDK Help.
 you would select *ScreenX Property*.
 8. Apply the character style **Topiclink**. This displays the topic text in italics.
 9. With the text still selected, select Special > Hypertext.
 10. In the Hypertext dialog, select **Go to URL** in the **Command** drop-down list.
 11. In the **Syntax** field, do the following:
 - Type Msi.chm:./setup/ after the text *message URL*, leaving a space after *URL*.
 - Paste the topic name after the final /.
 - type .htm after the topic name.
 When you finish, the contents of the **Syntax** field should look like this:
 message URL Msi.chm:./setup/screenx_property.htm
 12. Click the New Hypertext Marker button.
- When help is created, the marker is converted to a hyperlink.

Linking to a Topic in Visual Studio .NET

 *Wise for Windows Installer only.*

Use this procedure to embed a link to a topic in the Visual Studio .NET help collection. For guidelines on the form to use, see [Cross References to a Third-Party Help System](#) on page 44.

Note

Follow these instructions exactly. Sometimes a hyperlink will look correct in the Frame file but in the help, only the first character of the hyperlink text is formatted as a hyperlink. The only thing we can think of is that the steps for applying the character tag and adding the marker were not followed precisely. If this happens, it is best to remove the character tag, remove the marker, and start over.

1. In Frame, type the reference in the text but do not apply any special formatting yet.
 Example:
 See ScreenX Property in the SDK Help.
2. Open Visual Studio .NET and find the topic to reference.
 At the top of the help window is a drop-down list that contains the topic pathname.
3. Click in the drop-down list and make sure the entire pathname is highlighted.
 Example:
 ms-help://MS.VSCC/MS.MSDNVS/msi/pref_7qnn.htm
4. Copy the contents of the drop-down list to the clipboard.
5. In Frame, select the topic title in the text. Example: If the text says:
 See ScreenX Property in the Visual Studio .NET Help.
 you would select *ScreenX Property*.
6. Apply the character style **Topiclink**. This displays the topic text in italics.
7. With the text still selected, select Special > Hypertext.
8. In the Hypertext dialog, select **Go to URL** in the **Command** drop-down list.

9. In the **Syntax** field, paste the topic name after the text *message URL*, leaving a space after *URL*.

When you finish, the contents of the **Syntax** field should look like this:

message URL ms-help://MS.VSCC/MS.MSDNVS/msi/pref_7qnn.htm

10. Click the New Hypertext Marker button.

When help is created, the marker is converted to a hyperlink.

Using Conditional Text

We use FrameMaker's conditional text feature to show or hide certain elements from the various outputs. When you open a documentation project, change the book's conditional text settings to show all conditional text. Leave the conditional text on until the end of the documentation project, then show or hide it as appropriate just before producing the final output.

To change conditional text settings for multiple documents or the entire book, open the book file, select the documents to affect, and select View menu > Show/Hide Conditional Text.

The documentation template contains the conditional text tags that are common to both Altiris and Wise products. (See <<x-ref to template>>.) Each product line's documentation can have additional condition tags.

Wise documentation uses conditional text extensively for single-sourcing. See [Condition Tags in Single-Sourced Documents](#) on page 128.

Also see [Conditional Text Shortcuts](#) on page 111.

How to Use the not for help Tag

<<update or omit>>

Typically, we only include screen shots in the help when they contain callouts. That means we must delete unwanted images before converting the Frame files to help. A time-saving way to do this is to apply a conditional text tag to an image that should not be included in help, and then hide that tag before the help conversion.

Use the **not for help** condition tag to hide images that you don't want to appear in help. Text that has this condition tag is displayed in cyan. However, anchored frames do not change color, so it will be hard to see this condition tag at a glance. The return character (¶) that follows the image should be in cyan, but you can't always see it.

Only use the **not for help** condition tag for images. If you apply it to a paragraph, that paragraph is displayed in cyan, and will appear as cyan in the .PDF.

To hide an image from help:

1. Select the anchored frame AND the return character (¶) that follows it. If you do not select the return character, an extra blank line will appear in the help topic.
2. Select Special > Conditional Text.
3. Apply the condition tag **not for help**. The image does not change color; only the return character appears in cyan.
4. Before you convert the Frame files to help, you hide the **not for help** condition tag. This is part of the help creation checklist.

Also see [Using Conditional Text](#).

Conditional Text Shortcuts

To	Use this shortcut
Apply a condition tag to selected text.	<ul style="list-style-type: none">◆ Ctrl+4◆ Type the first characters of the tag.◆ Press Enter.
Make the selected text unconditional.	Ctrl+6
Display the Show/Hide Conditional Text dialog.	Alt+s, t
Turn condition indicators on or off.	Esc v Shift+o
Select all text around the insertion point that has the same condition tag settings.	Esc h Shift+c

Inserting Temporary Author's Notes

As you write, you might want to insert a temporary note to yourself, another writer, or a reviewer. Use triple greater-than/less-than brackets <<<like this>>> to enclose temporary notes that should NOT appear in the final document or Frame file. It's easy to search for and eliminate these comments before preparing the final book.

To insert a note that does not appear in the final user document, but that should remain in the Frame file, use the comment conditional text tag in FrameMaker. See [Using Conditional Text](#) on page 110.

Multilingual Production Requirements

- ◆ Leave 30% space in headings so that the heading won't go over 2 lines.
- ◆ Review crowded graphics (examples: WiseUpdate and WebDeploy).
- ◆ Make sure that there is space in all graphic callouts to add 30% more text.
- ◆ Don't repeat a graphic's callouts in text.
- ◆ If a graphic is in the book because it visually shows an example of what to type in a dialog, delete the graphic and integrate the example into the body text.

Chapter 6

Wise Documentation Sets

This section describes the pieces that comprise the documentation set for each Wise product. It is subject to change with each new product release. For information on documentation sets for older products, see the file named Documentation Set.doc in each product's subdirectory of Technical Publications\Archives\.

For information on where the Wise documentation files are stored, see [File Locations](#) on page 121.

Topics include:

- ◆ [Wise Package Studio](#).
- ◆ [Wise for Windows Installer](#).
- ◆ [Wise for Visual Studio .NET](#).
- ◆ [Wise Installation System](#).
- ◆ [Wise InstallTailor](#).

Wise Package Studio

Current version = 5.5.

The Wise Package Studio documentation set consists of several books. Currently, we do not offer printed manuals for any of the Package Studio books except the Getting Started Guide.

All files for .PDFs and printed manuals were created in FrameMaker 7.0. All online help files were created with Quadralay AutoMap 2003, using the Frame files as sources.

Deliverable	PDF - product	PDF - quick printing	PS - hi-res printing	Help	Other
Application Gateway				HTML	
ConflictManager Reference	X	X		CHM	
Debugger standalone help				CHM	
Management Reports help				HTML	
Release Notes					HTM
Software Manager Reference*	X	X		CHM	
WIE Inline Help					HTM
Windows Installer Editor Reference*	X	X		CHM	
Wise Package Studio Getting Started Guide	X	X	X		
WiseScript Editor Reference*	X	X		CHM	
WiseScript Express Reference*	X	X		CHM	
Workbench Inline Help (tools & processes)					HTM
Workbench Reference*	X	X		CHM	
Installation Notes (help)					HTML
Quality Assurance Getting Started	X	X	X		
Quality Assurance Reference*	X	X		CHM	
Release Notes (QA)					HTM

* Single-sourced book.

Wise for Windows Installer

Current version = 6.0.

The Wise for Windows Installer documentation set consists of several books. We offer printed manuals for all of the WfWI books.

This release includes the functionality that was formerly available in the separate product, Wise for Visual Studio .NET.

All files for .PDFs and printed manuals were created in FrameMaker 7.0. All online help files were created with Quadralay AutoMap 2003, using the Frame files as sources.

Deliverable	PDF - product	PDF - quick printing	PS - hi-res printing	Help	Other
Debugger standalone help				CHM	
Release Notes					HTM
Software Manager Reference*	X	X	X	CHM	
WfWI Getting Started Guide	X	X	X		
WfWI Inline Help					HTM
Wise for Windows Installer Reference*	X	X	X	CHM	
WiseScript Express Reference*	X	X	X	CHM	

* Single-sourced book.

Wise for Visual Studio .NET

As of version 6.0, this functionality is included with the Wise for Windows Installer product. Therefore, we no longer produce separate documentation for Wise for Visual Studio .NET. See [Documenting the Visual Studio Integrated Editor](#) on page 137.

The following information applies to the last version of WfVS as a separate product, 5.2.

The Wise for Visual Studio .NET documentation set consists of several books. We do not offer printed manuals for any of the WfVS books except the Getting Started Guide.

All files for .PDFs and printed manuals were created in FrameMaker 7.0. All online help files were created with Quadralay AutoMap 2003, using the Frame files as sources. The main help for WfVS is converted to Microsoft Help format and integrated with the Visual Studio .NET help collection.

Deliverable	PDF - product	PDF - quick printing	PS - hi-res printing	Help	Other
Download Redistributables standalone help				CHM	
Release Notes					HTM
Software Manager Reference*	X			CHM	
WfVS Getting Started Guide	X	X	X		
Wise for Visual Studio .NET Reference*	X			HxS	
WiseScript Express Reference*	X			CHM	

* Single-sourced book.

Wise Installation System

Current version = 9.0.

With the release of 9.0, this product consists of Wise Installation System - Standard Edition (formerly InstallBuilder) and Wise Installation System - Professional Edition (formerly InstallMaster). The InstallMaker product has been discontinued.

Note

Most changes that are made to the Wise Installation System documentation must also be made to the corresponding version of WiseScript Editor, and vice versa.

- ◆ Wise Installation System Reference Manual - printed (Frame 6.0)
- ◆ Wise Installation System Reference Manual - .PDF (Frame 6.0)
- ◆ Wise Installation System Getting Started Guide (Frame 6.0)
- ◆ Wise Installation System Online Help - contains all info in Reference Manual (RoboHelp 9)
- ◆ Wise Installation System Release Notes - .HTM

WiseUser 9.0

This product is an optional add-on to the Wise Installation System.

- ◆ WiseUser Reference - .PDF only (Frame 6.0)
This contains both installation instructions and reference material for using WiseUser.

Wise InstallTailor

Current version = 6.0.

This is a stand-alone version of the InstallTailor tool that is in Wise Package Studio.

The only deliverable is a help file, InstallTailor.chm.

The help is created from a separate Frame document, which consists of some introductory material plus versions of the 2 InstallTailor topics from the WPS manual.

Chapter 7

File Management

This section discusses <<<>>>.

Topics include:

- ◆ [File Locations.](#)
- ◆ [Sharing Files.](#)
- ◆ [Archiving Documentation Files.](#)

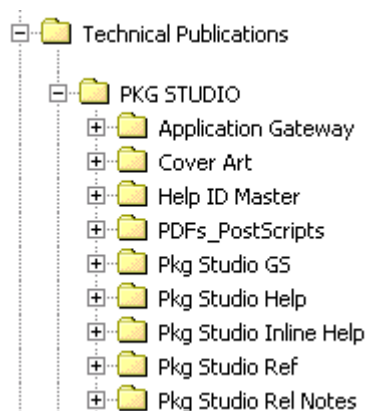
File Locations

The source files for all Wise documentation as well as the files that actually shipped with the product are on the Wise Solutions Departments server under the Technical Publications directory. The server is backed up nightly.

Working Files

Each current product has its own directory and subdirectories within Technical Publications. These are the working files for the current release.

Example: The directory structure for the current release of Wise Package Studio might look like this:



Note

Read any Readme files located in the source file directories.

Archive Files

The Archives directory within Technical Publications contains documentation for previous product releases. If you need to refer to an older document, you can open it from the archives, but *do not change anything in the archives*. See [Archiving Documentation Files](#) on page 124.

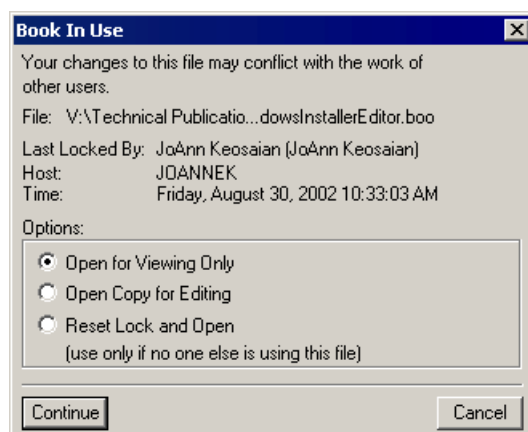
Administrative Files

The Archives directory within Technical Publications contains department files. Examples: procedures, templates, utilities, management spreadsheets, and so on.

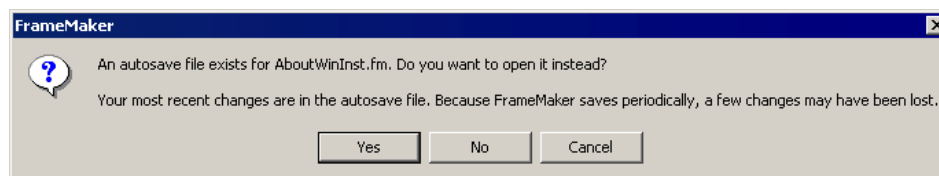
Sharing Files

We do not use a source code control system for documentation files; instead, we rely on communication and organization to maintain file integrity. FrameMaker displays an error message when you try to open a file that another writer is using. If you notice any of the indicators below when opening a file, then another writer might be working in the file.

Indicators From FrameMaker That a File is In Use



If you see the “In Use” message above, click Cancel and verify with the other writer that the file is actually in use. Sometimes, this message appears erroneously. If you verify that the file is not in use by the other writer, and the above message still appears, click **Reset Lock and Open** and click Continue.



If you see the “Autosave file exists” message above, click Cancel and ask all other writers if the file is in use. If you verify that the file is not in use by any other writers, and you still get the above message, then click Yes in this dialog to use the Autosave file.

If you open a file and see NO error messages, but the file seems to be non-editable, then the file might be in use by another writer. Check with all other writers.

General Guidelines

- ◆ Close all book and FrameMaker files when you are not actively using them, even if you are just going to lunch or a meeting.
- ◆ When you are assigned to work on a specific book or chapter, you usually “own” the relevant files. Notify the other writers of the ownership so that they do not edit those files. When you finish with a file or set of files, notify whoever is coordinating the project so the other writers know they can access those files again.
- ◆ If you need to edit a file that another writer owns, ask their permission before editing the file or wait until they complete their changes.
- ◆ If you need to update a book, check with the other writers who are working on the same book to make sure their files are closed. Otherwise you will receive errors.

- ◆ (RoboHelp only.) DO NOT open a help project that another writer has open. Even though RoboHelp lets you do this without providing any error messages, it corrupts the project file. Typically, only one writer is assigned to a specific help project so this should not happen often. However, ask first if you're not sure.

The bottom line is, when in doubt, ask.

Archiving Documentation Files

Archiving to the Network

Documentation archives are located in the Archives subdirectory in the Technical Publications directory on the network.

At the end of every release, archive all files related to that release.

1. Clean out all folders. Remove all temporary, test, and junk directories and files. Delete any backup directories (example: help backups).
2. If you haven't done so for this release, rename all directories and check for broken image references. See <<<I think there's a topic that covers this>>>
3. Copy the entire project directory and its subdirectories to the Archives directory.

Note

For Wise Package Studio, first make a directory in Archives named Pkg Studio x.y All Parts. Then copy the Package Studio, ConflictManager, WIE, WSE, and any other component directories there.

4. In each archive directory that contains Frame files (example: the GSG and Ref directories), delete all backup Frame files. They take up space on the drive and are not needed. Use Windows Explorer's Search feature and search for files named *backup*.
5. Delete all chapter-level TOC files, such as WelcomeTOC.fm.

Caution

Do not delete the book TOC file.

Archiving to CD

In addition to archiving on the network, we also archive to CD periodically. This safeguards against loss of network data. It also lets us free up network disk space by deleting old files. A list of files that are on CD archives but are NOT in the Technical Publications directory is located in V:\Technical Publications\Archives\AAA Readme.

- ◆ Periodically, copy archived documentation directories to CD. You can do this every few releases.
- ◆ Organize the CDs by a single product or by related products.
- ◆ Make 2 copies of each CD. Keep one in the department and one off-site at the Technical Publications Manager's house.
- ◆ Also make periodic copies of the Admin and Misc subdirectories in the Technical Publications directory.

Also see:

[Archiving Single-Sourced Books](#) on page 127

[Archiving Documents That Contain Text Insets](#) on page 136

Chapter 8

Single Sourcing

In 2003, we began single-sourcing some of the Wise documentation, using a single Frame book file to produce deliverables for multiple products. Example: The version of Software Manager that is shipped with Wise for Windows Installer is somewhat different from the version that is shipped with Wise Package Studio. We use conditional text, variables, and text insets to manage the differences between the versions.

Topics include:

- ◆ [Planning Conditional Documents.](#)
- ◆ [Managing Single-Sourced Books.](#)
- ◆ [Condition Tags in Single-Sourced Documents.](#)
- ◆ [Rules for Tagging Conditional Text.](#)
- ◆ [Product Name Variables.](#)
- ◆ [Text Insets.](#)
- ◆ [Documenting the Visual Studio Integrated Editor.](#)

Planning Conditional Documents

This information was copied from *Planning Conditional Documents* in the FrameMaker help and modified for our purposes.

- ◆ When planning a conditional document, examine the nature of the material and, because others may have to work in or maintain a conditional document you create, look at how several people can take turns working with it. Plan to treat conditional material consistently to make the document easier to use and maintain. Others will need to know the conventions you followed as you applied condition tags. (See [Rules for Tagging Conditional Text](#) on page 128.)
- ◆ Number of versions. Define how many versions your finished project will contain. Example: If you're creating a manual that describes a program that runs on both Windows and UNIX platforms, you might want to be able to produce at least 2 versions: a Windows version and a UNIX version. To produce these versions with editorial comments sprinkled in the text during the review process, you'll need even more versions: UNIX with comments, UNIX without comments, Windows with comments, and Windows without comments.
- ◆ Number of condition tags required. Decide how many condition tags you need to produce the desired versions. A version of a document is defined by a unique set of condition tags. Example: A version of a finished Windows manual might be defined by having a Windows condition tag showing, a UNIX condition tag hidden, and a Comments condition tag hidden. In this example, you would need to decide whether to use one condition tag for Windows comments and another for UNIX comments, or whether to use a single condition tag for both Windows and UNIX comments.
- ◆ Organization of content. Evaluate the extent to which the document can be conditional and how you can organize the material to simplify development and maintenance. Example: You might be able to organize a book so that conditional text is limited to a few documents. Or you might choose to keep versions of a particular chapter in separate files rather than in conditional text, and then use a different book file for each version of the book.
- ◆ Variables. Words and phrases that are used repeatedly in a document (example: product and company names) are easier to work with as variables than as conditional text.

Decide whether your conditional documents will need different variable definitions for each version. You can create a template for each version of the document. Each template should have only one version visible and should define the variables for that version. You can then use File > Import > Formats to switch the variable definitions from one version to another. See [Product Name Variables](#) on page 132.

Managing Single-Sourced Books

- ◆ For each version of the book, decide which conditions to show or hide.
- ◆ Whenever you edit the book, show all conditional text and show condition indicators.
- ◆ Always leave files with the Show/Hide settings set to Show All, and the Condition indicators shown. Otherwise, someone editing a different book might encounter broken cross-references to a hidden section of your book.
- ◆ When printing a single-sourced document for review, print each version of the book.
- ◆ When searching for unresolved cross-references during the finishing steps, first show all conditional text. Then show only one version at a time and search again.
- ◆ When performing a spell check during the finishing steps, do a separate spell check operation for each version of the book. This will help find problems with word spacing caused by incorrect condition tagging.
- ◆ When you make a .PDF or PostScript of a book, check conditional text settings *every time*. Example: Suppose you make a .PDF of the WfWI version of Software Manager and then you make a .PDF of the WPS version. A few days later, you need to remake the WfWI version. Even if you think you remember what state the book is in, you must always check because someone else might have changed the settings. Use condition templates to facilitate changing conditional text settings. See [Changing the Book View](#) on page 158 and [Creating Condition Templates](#) on page 131.
- ◆ When you create help, you do not need to change the Show/Hide settings in the Frame book because they are specified in WebWorks.
- ◆ Maintain a single Map ID file for single-sourced books.

Also see [Archiving Single-Sourced Books](#).

Archiving Single-Sourced Books

After you complete the finishing steps and the product is released, archive the single-sourced book files with each product that was included in the release.

Examples:

- ◆ If Wise releases version 6.0 of WfWI, but there is no corresponding release of WIE, archive the Windows Inst files with WfWI only.
- ◆ If Wise releases WfWI and WPS at the same time, archive a copy of Software Manager for each of those products.

Note

Up to July 2003, we archived the Software Manager and WiseScript Express books under their own directories in the Archives directory.

Also see [Archiving Documentation Files](#) on page 124.

Condition Tags in Single-Sourced Documents

We use conditional text to manage the differences between multiple versions of a book. We also use conditional text in the FrameMaker files to hide certain elements from the book and help. See [Using Conditional Text](#) on page 110.

See:


[Rules for Tagging Conditional Text](#) on page 128


[Creating a New Condition Tag](#) on page 129

[Creating Condition Templates](#) on page 131

Rules for Tagging Conditional Text

- ◆ If you are using conditional text in one chapter of a book, then all chapters must contain that condition tag. Otherwise, when you show/hide text for the entire book, that tag will not be listed.
- ◆ The smallest unit of conditional text is one sentence. This facilitates translation.
- ◆ If the conditional text begins or ends with punctuation, make the punctuation conditional too.
- ◆ To avoid word spacing problems, always include the space after a sentence in the conditional text, except at the end of a paragraph.
- ◆ Be very careful about including the paragraph mark (§) in conditional text.
 - Include the paragraph mark if the entire paragraph has the same condition tag.
 - Do not include the paragraph mark if the paragraph contains more than one condition tag, or if part of the paragraph is unconditional.
- ◆ Do not put conditional text on the same paragraph line as a table, especially if the table is unconditional.
- ◆ Apply conditional text in the same order throughout the document. Our convention is to apply the conditional tags in alphabetical order. In the example below, the condition tags have been applied in this order: WfWI only, WPS.

Text for WfWI only.  Custom report information is stored in the file Report.ini, which is located in the Reports directory specified in Wise Options. Custom report information is stored in the file Report.ini, which is located in the Reports directory specified in Workbench Preferences.¶

Text for WPS only.  Custom report information is stored in the file Report.ini, which is located in the Reports directory specified in Workbench Preferences.¶

- ◆ Do not apply more than one condition to text, because when text has more than one condition tag, the text will be visible when *any* of its tags are shown. Text that has multiple condition tags is displayed in magenta, so it should be easy to see when this happens.

Tagging Anchored Frames

- ◆ If an anchored frame and its entire contents are conditional, then you can apply the condition tag to the anchored frame. Also select the paragraph mark that the frame is anchored to, otherwise you'll end up with an empty paragraph when that condition is hidden. You then do NOT have to apply the condition tag to every callout within the frame.
- ◆ If the anchored frame contains an image and callouts, and the image is conditional, then you must repeat the anchored frame and change its contents accordingly for

each situation. Then apply the condition tag to the entire frame and its paragraph as above.

- ◆ If only some of the callouts in an anchored frame are conditional, then apply the condition tag only to that specific text.

Tagging Tables

When tagging tables, the same principles apply as for anchored frames:

- ◆ If the entire table is conditional, tag the paragraph marker that the table is anchored to.
- ◆ If only certain contents are conditional, tag the contents.

Tagging Help Navigation

Sometimes a series of help navigation cross-references (that typically appear at the end of a topic) contains a reference that must be made conditional. In such cases, do not apply the help navigation condition to the cross-references. Apply the appropriate condition to the cross-reference that must be conditional. This is because applying more than one condition tag to any text produces undesirable results.

Creating a New Condition Tag

Before you create a new condition tag, discuss the change with the Technical Publications Manager.

1. In *In Frame*, open a chapter of the book to which you're adding the condition tag.
2. Define the color for the new condition tag. For details, see *Defining and modifying colors and tints* in the FrameMaker help. Also see [Conditional Text Colors](#) on page 130 for color suggestions.
3. Select **Special > Conditional Text**.
4. Click **Edit Condition Tag**.
5. In the **Edit Condition Tag** dialog:
 - In **Tag**, enter the name of the new condition. Typically, this should be an abbreviation of the product name (example: WfWI only, or WfWI+WIE).
 - Leave **As Is** in the **Style** drop-down list.
 - From **Color**, select the new color you defined.
 - Click **Set**.
6. Save the file.
7. Make the book file active and select all files in the book, except the one you added the condition tag to.
8. Select **File > Import > Formats**.
9. In the **Import Formats** dialog,
 - From **Import from Document**, select the chapter you added the condition tag to.
 - Mark **Color Definitions** and **Cross-Reference Formats**.
 - Clear all other options.
 - Click **Import**.
10. Save and close the open chapter and the book.
11. Document the new color and condition in the Doc Procedures.

Conditional Text Colors

This section describes the colors used in our conditional text tags. Technically, the available colors are unlimited, however, when displayed in type next to each other, many colors are indistinguishable. Therefore, we have limited colors to work with. If you need to add a condition tag, experiment with the color until you find one that can be distinguished from the tags with which it will be used.

Note

Conditional text appears in magenta when it is assigned 2 or more overlapping condition tags that use different colors as condition indicators. Therefore, do not use magenta to indicate a particular condition.

Before you create a new condition tag, you must first create a new color definition. See *Defining and modifying colors and tints* in the FrameMaker help.

Color Settings for Conditional Text Tags Currently in Use

Color values are expressed in C-M-Y-K percentages.

Condition Tag	Color Name	CMYK Value
comment	Gray	C-39, M-27, Y-26, K-2
Exp+NS	Brown	C-10, M-36, Y-62, K-31
Express	DkBlue	C-98, M-52, Y-7, K-29
help navigation	Green *	C-100, M-0, Y-100, K-0
MgtReports	BrightPurple	C-30, M-70, Y-0, K-15
MSI+WIE	DkBlue	C-98, M-52, Y-7, K-29
MSI_Editor	BrightPurple	C-30, M-70, Y-0, K-15
not for help	Cyan *	C-100, M-0, Y-0, K-0
standalone help	As Is (black)	
QA	Brown	C-10, M-36, Y-62, K-31
WfWI only	Red *	C-0, M-100, Y-100, K-0
WfWI+WIE	Mulberry	C-16, M-100, Y-53, K-0
WIE only	Aqua	C-75, M-0, Y-0, K-25
WIS	MidGreen	C-85, M-0, Y-50, K-24
WIS+WSE	Red *	C-0, M-100, Y-100, K-0
WS-NS	Purple	C-43, M-91, Y-0, K-0
WSE	DkGold	C-20, M-33, Y-78, K-0
WPS	DkGreen	C-99, M-0, Y-60, K-45

* denotes a predefined color in FrameMaker.

Unused Colors

These colors will work with the existing conditional text colors.

Color Name	CMYK Value
Orange	C-0, M-46, Y-100, K-0
Lavender	C-32, M-32, Y-0, K-0
Beige	C-0, M-12, Y-37, K-0 (might be too hard to see)

Creating Condition Templates

To set conditions for a specific version of a book, you use a special template that contains the appropriate condition settings and variable definitions.

If condition templates exist:

You can use condition templates from a previous release, if condition tags have not been added or changed to the master template since the condition templates were created.

To create a new condition template:

1. Open a short chapter in the single-sourced book; the Front Page or Welcome chapter will do.
2. Select File > Save As and give the file a new name.
 - Store condition templates in a subdirectory of the main book directory. Example: Windows Inst\Condition Templates.
 - Append _template to the file name. Example: WfWIRef_template.fm, or SoftwareMgr_WfWI_template.fm.
3. Change the conditional text settings as follows:
 - Select Special > Conditional Text.
 - In the Conditional Text dialog, click Show/Hide.
 - Change the Show/Hide settings for this version of the book. Refer to the table in [Hide Conditional Text](#) on page 157.
 - Clear the **Show Condition Indicators** checkbox; this changes all the conditional text colors to black.
 - Click Set.
4. If variables are used, define them. See [Product Name Variables](#) on page 132.
5. Save the template file.
6. Repeat these steps for each version of the book.

For information on using the condition templates, see [Changing the Book View](#) on page 158.

Product Name Variables

Single-sourced books use a variable for the product name. When you create a new single-sourced book that meets the following criteria, create a unique variable for the product name:

- ◆ If it must be combined with other books in a documentation set (example: Wise Package Studio).
- ◆ If any other books in the doc set use a variable for the product name.
- ◆ If the value for the product name in any other book is different from the one in the new book.

Product Name Variables Currently in Use

Book	Variable Name	Possible Values
Software Manager	Product_SM	<ul style="list-style-type: none"> ◆ Wise for Windows Installer ◆ Wise Package Studio
<ul style="list-style-type: none"> ◆ WindowsInst ◆ WindowsInstallerEditor 	Product_WinInst	<ul style="list-style-type: none"> ◆ Windows Installer Editor ◆ Wise for Windows Installer
<ul style="list-style-type: none"> ◆ Express ◆ WiseRef ◆ WiseScriptEditor ◆ WiseScriptforNS 	Product_WiseScr	<ul style="list-style-type: none"> ◆ Wise Installation System ◆ WiseScript Editor ◆ WiseScript Express ◆ WiseScript for NS
<ul style="list-style-type: none"> ◆ Express ◆ WiseRef ◆ WiseScriptEditor ◆ WiseScriptforNS 	ScrEditorUI	<ul style="list-style-type: none"> ◆ Script Editor ◆ Setup Editor
<ul style="list-style-type: none"> ◆ Workbench ◆ QualityAssurance 	Product_WB	<ul style="list-style-type: none"> ◆ Quality Assurance ◆ Wise Package Studio

Text Insets

A text inset is a FrameMaker file that can be reused in multiple documents. To add the text inset to a document, you import the inset file by reference. When you import a text inset, Frame automatically stores the reference as a relative path. As long as the document remains in the same location relative to the text inset, the document is updated automatically when the text inset is changed.

Use a text inset when you need to add the same text to multiple places within a book, within a product's documentation set, or to multiple books across product lines. The text must be generic enough to work in all situations. A text inset that is used only within one book can be more specific than a global text inset that is used across product lines.

See:

[Text Inset Guidelines](#)

[Creating a Text Inset](#) on page 134

[Importing a Text Inset](#) on page 134

[Updating Text Insets](#) on page 136

[Archiving Documents That Contain Text Insets](#) on page 136

Text Inset Guidelines

<<<add more guidelines as we find them>>>

- ◆ A text inset can be a single sentence or an entire topic.
- ◆ A text inset always ends with a paragraph marker. Therefore, you cannot place a text inset at the beginning or middle of a paragraph in the target document.
- ◆ Index markers work in text insets. Make sure the index entry adheres to standards and is compatible with the index in the target document.
- ◆ When you create a help project for a book, and someone changes a text inset while the book is open in WebWorks, the text inset is not updated in the help unless you open and re-save the Frame file containing the text inset, and then regenerate the help.
- ◆ Text in text insets is not included in Frame search or spell check operations.
- ◆ Variables work in text insets. Make sure the document that uses the text inset has a definition for the variables used.
- ◆ When a text inset contains change bars, they appear in the target document. When you clear change bars in the target document, they are cleared in the imported text inset. However, they are NOT cleared in the source text inset. Therefore, do not turn on change bars in a text inset file.

Cross-references

- ◆ Within a text inset, do not use cross-references to other topics outside the text inset. They seem to work in the help but definitely do not work in the .PDF. Also, it would be too hard to fix broken ones.
- ◆ To cross-reference from the target file to material in a text inset, it is best to type the heading in the target document and not in the text inset.

It is possible to cross-reference to a heading in a text inset. However, the cross-reference is easily broken when the text inset is changed. Sometimes the cross-reference becomes unresolved, other times it remains resolved but the cross-reference text changes to something you don't want.

- ◆ Cross-references from within a text inset to other topics within the same text inset DO work.

Conditional Text

- ◆ Applying conditional text tags *within* a text inset works, provided the target documents contain the same condition tag definitions.
- ◆ When a paragraph containing a text inset has a conditional text tag, and you update the text inset source, the condition tag is removed from the target document. Therefore you have to find those instances and re-apply the condition tags. This has been reported to Adobe as a bug, but so far there is no workaround. Use caution when applying conditional text to text insets. See [Condition and Style Problems With Text Insets](#) on page 135.

Creating a Text Inset

Where to store text insets:

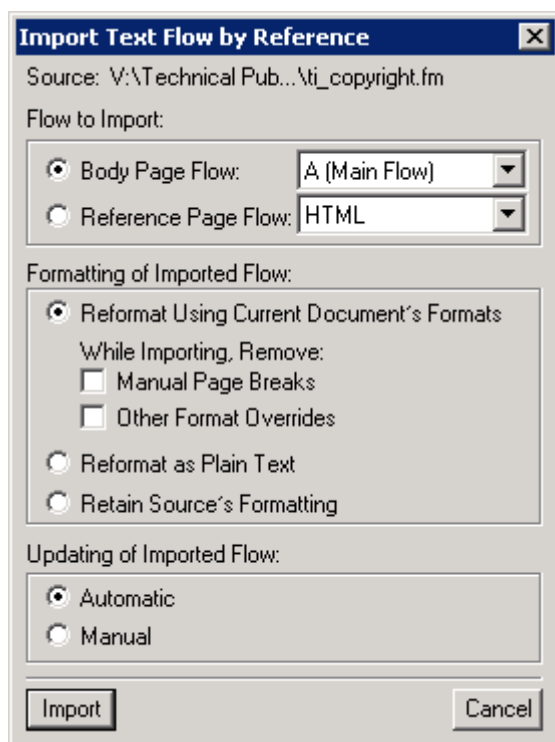
- ◆ If the text inset can be used by multiple books and products, store it in Technical Publications\Text Insets.
- ◆ If the text inset is specific to one product, or to one feature that is used by several products, store it in a subdirectory of Technical Publications\Text Insets. Examples:
 - Text Insets\Reports
 - Text Insets\SoftwareManager

To create a text inset:

1. Open ti_template.fm, which is in Technical Publications\Text Insets.
2. Type or paste the text. It can be a single paragraph or an entire topic.
3. Format the text using standard style tags.
4. Do a spell check. This is important because Frame's spell-check does not operate on text insets in a document.
5. Select File > Save As and specify a file name, using the convention ti_<descriptive name>.fm. Specify the location according to the guidelines above.

Importing a Text Inset

1. In a Frame file, click where you want to insert the text inset.
2. Select File > Import > File.
3. In the Import dialog, select the text inset file, make sure the **Import by Reference** option is marked, and click Import.
4. In the Import Text Flow by Reference dialog, accept the defaults as shown in the screen shot below and click Import.



The text inset is imported to the document. When you click anywhere in the text, the entire text inset is selected. You cannot edit the text within the document; you must edit the source text inset.

5. If the paragraph following the text inset has no text (usually when the text inset is the last paragraph in the document), add a nonbreaking space (Ctrl+space) to the paragraph following the text inset. (This is per Adobe Support Knowledgebase document 315339.)

When you update a text inset, the paragraph tag and properties of the text inset are applied to the paragraph that follows the inset. For us, this is mainly a problem when the inset is the last paragraph in a topic. It is especially problematic when the last paragraph in a topic is a text inset *and* it's conditional. Then, the cross-ref marker at the beginning of the next heading picks up the condition that's applied to the text inset, which will probably result in broken cross-refs when you hide that condition.

6. If you need to make the imported text inset conditional, or if you need to apply a different paragraph tag, see [Condition and Style Problems With Text Insets](#).

Condition and Style Problems With Text Insets

When a paragraph containing a text inset has a conditional text tag, and you update the text inset source, the condition tag is removed from the target document. This has been reported to Adobe as a bug, but so far there is no workaround.

The same thing happens when you apply a different paragraph tag to an imported text inset; when you update the text inset source, the style tag is removed in the target document.

When you apply a condition or style tag to an imported text inset:

1. Insert a comment above the text inset. Be sure to use the phrase “conditional inset.” Examples:

```

{{{conditional inset: make the following inset WPS}}}
{{{conditional inset: make all insets in this topic WFWI only}}}
{{{conditional inset: make the following inset B Body}}}

```

2. Apply the comment conditional text tag.

When you update a text inset:

1. Search the book or books that include that inset for “conditional inset”.
2. Re-apply the proper conditional text tag or paragraph tag.

Updating Text Insets

<<<Need to develop guidelines for updating text insets>>>

Archiving Documents That Contain Text Insets

When you copy a document to the Archives folder, or when you copy change bar documents to a folder for the curriculum developer, the text insets in the document become unresolved and are no longer updated when the text inset file is changed. This is the desired behavior; we don’t want archived files to change.

If you open an archived document that contains text insets, an unresolved text inset message appears. Click OK. The file opens and the text inset is not updated.

If you need to edit an archived document that contains a text inset, and the text you need to edit is within the text inset, convert the text inset to editable text. You will rarely need to do this.

To convert a text inset to editable text:

1. In the Frame file, click anywhere in the text inset.
Select Edit > Text Inset Properties.
2. Click the Convert to Text button.
3. In the Convert to Text dialog, select an option to convert the select text inset or all text insets and click the Convert button.
The text inset is converted to editable text.

Also see [Archiving Documentation Files](#) on page 124.

Documenting the Visual Studio Integrated Editor

In release 6.0, the Wise for Visual Studio .NET functionality was integrated into the Wise for Windows Installer product. The WfWI and WfVS functionality are now covered in one book.

Product Name

Refer to the product as Wise for Windows Installer. Examples:

- ◆ Wise for Windows Installer is an authoring environment for Microsoft's Windows Installer service.
- ◆ When you use the Resources page to add binary resources to an installation, Wise for Windows Installer adds the appropriate rows to the Binary table.
- ◆ Each time Wise for Windows Installer scans a solution, it adds new files to the installation. [Although the files are added within the Visual Studio integrated editor, it's the product—WfWI—that does the scanning.]

Interface Names

Refer to the 2 interfaces as Wise editor and Visual Studio integrated editor. Note the case. Note the use of articles in the examples:

- ◆ Clear this if you don't want to copy the installation file (recommended in the Wise editor only).
- ◆ When you work in the Visual Studio integrated editor, you typically create an installation as a project within a Visual Studio .NET solution.
- ◆ Each installation project in the Visual Studio integrated editor has settings that control...
- ◆ (Wise editor only.) Select Edit menu > Preferences. On the Source Control tab, mark Enable source control and click OK.
- ◆ (Visual Studio integrated editor only.) To display help specific to the current area you are using or task you are trying to complete, click a link in the Visual Studio .NET Dynamic Help window.

Using Conditional Text

Any text or image related to the Visual Studio integrated editor must have the condition "WfWI only." VS-related elements should never appear in the WIE book.

When the interfaces behave differently

Write as though the Wise editor is the default and add the Visual Studio method in parentheses. You do not have to include the "integrated editor" in these situations. Examples:

- ◆ To set general options, select Tools menu > Options and click the General tab. (In Visual Studio: Tools menu > Options > Wise Options > General.)
- ◆ You can specify whether you create standard Win32 or .NET installations, and customize how Wise for Windows Installer handles the .NET assemblies. Select Tools menu > Options and click the .NET Assemblies tab. (In Visual Studio: Tools menu > Options > Wise Options > Assemblies.)
- ◆ Click Distribute. (In Visual Studio: select Project menu > Package Distribution.)

When should you include the verb in the parentheses? When the verb in the VS method is the same as the initial verb (first 2 examples above), do not include it within the parentheses. When the verb in the VS method is different (third example above), do

include it within the parentheses. Also include the second verb if it is needed for clarity.
Example:

When you create a new installation, select Predefined Templates > Web Application on the New Installation File dialog. (In Visual Studio: select Wise Setup and Deployment Projects > Web Application on the New Project dialog or Wise Files > Web Application File on the New File dialog.)

When the differences between the interfaces are too long to put in parentheses, make separate H4s or separate topics. Examples:

- ◆ To start a new installation in the Wise editor:
- ◆ To start a new installation in the Visual Studio integrated editor:

If the text discusses something that can't be done in the Visual Studio integrated editor, you might have to add a note or caveat. The standard format is:

Not available in the Visual Studio integrated editor.

Chapter 9

Producing Inline Help

Wise for Windows Installer and Windows Installer Editor contain inline help within the product UI, which Technical Publications is responsible for writing and producing.

The Workbench interface in Wise Package Studio also contains inline help. The Workbench inline help is produced with FrameMaker and WebWorks Publisher. See [Creating Workbench Inline Help](#) on page 198.

Topics include:

- ◆ [Updating WfWI/WIE Inline Help.](#)
- ◆ [Linking to Online Help From Inline Help.](#)
- ◆ [Linking to SDK Help From WfWI Inline Help.](#)

Updating WfWI/WIE Inline Help

Wise for Windows Installer and Windows Installer Editor contain HTML help pages within Setup Editor. These are NOT the same as the online help in HTML format that you produce from the book files.

We maintain a spreadsheet to help keep track of the inline help pages; see the document WfWI Inline Help.xls in V:\Technical Publications\Admin\Notes Memos + Misc.

Note

Currently, we use the same files for both WFWI and WIE. Keep this in mind as you write and edit, and be sure not to include any product-specific verbiage.

To update the inline help files:

1. Check out the inline help pages and the graphics links from Visual SourceSafe.
2. Open or create the inline help page in Notepad. Change or write the text as needed for the new release.

Caution

Do not use PageMill to edit these files, because it adds non-standard HTML code.

3. Add or update links to the WFWI help and SDK help. See [Linking to Online Help From Inline Help](#) on page 141 and [Linking to SDK Help From WfWI Inline Help](#) on page 142.
4. When the changes are completed, check the files into Visual SourceSafe.
5. Add any new information to the document WfWI Inline Help.xls in V:\Technical Publications\Admin\Notes Memos + Misc.

Linking to Online Help From Inline Help

Do this after the WFWI or WIE Map ID file (.H) has been created.

1. Determine how many links you need to add. Obtain that number of unused Map IDs from Development.
2. Open the inline help page in Notepad.
3. In the inline help page, type the line of text that will become the link.
4. Apply the tag:
``
here *map* is the Map ID number. Example: In the tag ``, the Map ID is 10509.
5. Open the book in Frame and find the topic to link to.
6. Copy the TopicAlias marker text. If the topic does not have a TopicAlias marker, create one. See [Mapping Help IDs](#) on page 184.
7. Open the project Map ID file (.H), or create a new text file with the extension .H. See [Mapping Help IDs](#) on page 184.
8. Create an entry for this topic and Map ID, using the syntax:
`#define Working_with_Summary_Items<tab>595`
where `<tab>` is a tab character.
9. Repeat these steps for any other links to the product help.
10. Save the Map ID file. If you created a new file, append its contents to the project's master Map ID file.
11. Save the inline help page.
12. When the changes are completed, check in the HTML files to Visual SourceSafe.
13. Add any new information to the document WfWI Inline Help.xls in V:\Technical Publications\Admin\Notes Memos + Misc.

Linking to SDK Help From WfWI Inline Help

In order to link to the SDK Help, you must have a current version of the Windows Installer SDK Help installed on your system.

1. Open the inline help page in Notepad.
2. In the inline help page, type the line of text that will become the link.
3. Add the tag:

```
<a href="*1 name.htm">
```

where *name* is the file name of the SDK Help topic. To find the file name:

- Open WfWI and go to the SDK Help.
 - Find the help topic to link to.
 - Click anywhere in the help topic.
 - Select View > Source. This opens the HTML source in a Notepad window.
 - In Notepad, select File > Save As. This displays the Save As dialog, with the file name displayed in the File Name field.
 - Copy the file name, except for the bracketed number at the end of the name.
Example: If user_policies[1] is the default file name, copy only user_policies.
 - Paste the file name into the <a href> tag, just to the left of the .htm extension.
Example:
4. Repeat the above steps for all other SDK links on the page.

Chapter 10

Wise Documentation Deliverables

When documentation changes have been through editing and review, you are ready to produce deliverables. For most products, these include the book(s) in both electronic and printed form, an online help system, and a release notes file.

Topics include:

- ◆ [Deliverables.](#)
- ◆ [Preparing Documentation for Review.](#)
- ◆ [Delivering Beta Documentation.](#)
- ◆ [Guidelines for ReadMes/Release Notes.](#)
- ◆ [Finishing Steps Checklist.](#)
- ◆ [Housekeeping.](#)

This section does not cover the procedures for creating an online help system; see [Creating Help With WebWorks](#) on page 180.

Deliverables

This section lists the items that Technical Publications delivers to Wise at the end of every project. It also discusses the items that Technical Publications needs to obtain from Wise's Graphic Design department.

Deliverables from the Writer to Wise

The following table provides a general overview of what we need to deliver to Wise when a project is completed. For each release, particularly one that involves multiple products, we maintain a spreadsheet to keep track of the specific deliverables. For an example, see one of the spreadsheets from previous releases in V:\Technical Publications\Admin\Estimates + Schedules.

Deliverable	Consists of	Deliver to
Reference Manual, Getting Started Guide, other books	.PDF file of each book	Development (via SourceSafe)
	PostScript file and hard copy	commercial printer
	FrameMaker source files	dept. archives
Online help	Online help file (.chm)	Development (via SourceSafe)
	WebWorks help project files	dept. archives
Release Notes	HTML file and links	Development (via SourceSafe), dept. archives
Inline Help pages (WfWI and WIE)	HTML files and links. These are the HTML pages that appear in the lower right pane in Setup Editor.	Development (via SourceSafe), dept. archives
Inline Help pages (WPS)	HTML files. These are the help pages that appear in Workbench.	Development (via SourceSafe), dept. archives

See the following procedures for producing these deliverables:

[Creating a .PDF File](#) on page 167

[Creating a PostScript File for Commercial Printing](#) on page 171

[Creating Help With WebWorks](#) on page 180

[Producing Inline Help](#) on page 139

[Creating Workbench Inline Help](#) on page 198

Deliverables from Wise's Graphic Design

Wise's Graphic Design department supplies cover graphics for both online and printed books, as well as logo graphics for release notes files. Coordinate deadlines with Graphic Design for each new project, so that they can provide the cover art when you need it.

Because we now have so many different books, it is best to provide Graphic Design a list of exactly what you need for each release. For an example of the kind of information you

need to provide, see one of the graphic requirements spreadsheets under Technical Publications\Admin\Estimates + Schedules.

Graphics for the Product .PDF

Wise's Graphic Design supplies a .PDF-format cover page for each product .PDF. After you make the book .PDF, you replace the first page (which is just a placeholder page) with the cover page .PDF you obtain from graphic design. See [Creating a .PDF File](#) on page 167.

Ask for:

- ◆ Front cover, 8.5" x 11", color, .PDF format

Graphics for the Printed Getting Started Guide

If a Getting Started Guide is being printed, Wise's Graphic Design creates PostScript files of the front and back book covers and the book title page. You and Graphic Design then coordinate delivery of the files to the commercial printer.

Ask for:

- ◆ Front & back covers, 11"x8.5", color, PostScript format, for saddle stitch binding
- ◆ Title page, 5.5"x8.5", black & white, PostScript format

Occasionally, we want to have a small quantity of books printed at a quick printer. You might do this if Customer Service runs out of manuals before a new print order arrives, or if you want a nice hard copy of a book that we don't normally print. In these cases, you need .PDF-format graphics. You cannot use the same ones as for the online book because they are the wrong size and they don't include the back cover. Ask for:

- ◆ Front & back covers, 11"x8.5", color, .PDF format, for saddle stitch binding
- ◆ Title page, 5.5"x8.5", black & white, .PDF format

For details, see [Creating a .PDF File for Quick Printing](#) on page 177.

Graphics for the Printed Book

If a reference manual is being printed, Wise's Graphic Design creates PostScript files of the front and back book covers and the book title page. You and Graphic Design then coordinate delivery of the files to the commercial printer.

Ask for:

- ◆ Front & back covers, 7"x9", color, PostScript format
- ◆ Title page, 7"x9", black & white, PostScript format

Occasionally, we want to have a small quantity of books printed at a quick printer such as Kinko's. You might do this if Customer Service runs out of manuals before a new print order arrives, or if you want a nice hard copy of a book that we don't normally print, such as the WPS books. In these cases, you need .PDF-format graphics. You cannot use the same ones as for the product .PDF because they are the wrong size and they don't include the back cover. Ask for:

- ◆ Front & back covers, 7"x9", color, .PDF format
- ◆ Title page, 7"x9", black & white, .PDF format

For details, see [Creating a .PDF File for Quick Printing](#) on page 177.

Graphics for the Release Notes

With WfWI 6.0 and WPS 5.5, we began producing release notes with no external graphics. This makes it easier to distribute the HTML files without having to distribute the graphic files as well.

Preparing Documentation for Review

Preliminary Review

Preliminary reviews are done during the development phase, as you complete documentation for a specific enhancement or bug fix. Print the pages related to the change, including change bars, and give them to the developer and the QA tester assigned to that feature. Give them a reasonable amount of time in which to return it, but do provide a specific due date. Follow up on the due date.

Final Review

The final review of the entire book is done at the end of the documentation phase.

1. Select reviewers.

The reviewers should include the product manager and one or more QA testers. You might also ask for volunteers from Technical Support. (They usually provide valuable feedback, but their time constraints prevent them from reviewing much material.)

2. Schedule the review.

Typically, as we near the end of a development cycle, we set milestones for final activities, including documentation review. When the doc review milestones are set, notify the reviewers so they can schedule their time. Notify the reviewers of any schedule changes.

3. Prepare the review documentation.

Leave in the change bars. You also can show comments, especially if they are directed at the reviewer. Show and hide conditional text as appropriate for the product being released. If multiple products are being released at once, prepare a separate book for each product; it's too hard for reviewers to read a single-sourced document, especially in hard copy where they can't see the condition colors.

Typically, we deliver the review doc in 8.5x11 .PDF format. If a particular reviewer requests a hard copy, print it double-sided on the main office printer.

4. Deliver the review documentation.

Send the .PDF to the reviewer, along with the [Documentation Review Guidelines](#). Remind them of the due date for returning comments.

5. Follow up.

The day before the due date, check with the reviewer to make sure they will return their comments in time. Depending on how tight the overall release schedule is, you might be able to give them an extension.

Documentation Review Guidelines

Thank you for helping with technical review of our documentation. Please follow the guidelines below so that you can be sure to provide the kind of feedback we need. If you have any questions, ask a member of the Technical Publications team.

Please return by <<<>>>

What to do

- ◆ Read the documentation.
 - If the book has been rewritten for this release, or if it's for a new product, read the entire book.

- If the book has been revised for this release, vertical change bars in the left margin indicate what has been changed. You can concentrate on the changed material first and then review the rest of the book if there's time.
- ◆ Use the software as you follow the written procedures. This is especially important if you are not familiar with the new features in the release.

What you're looking for

- ◆ Note inaccuracies and tell us what IS accurate.
- ◆ Note missing material.
- ◆ Note things that are hard to understand or not explained quite right.
- ◆ Provide as much information as you can, with the goal of enabling the writer to make the change without further research. (Comments like "This is wrong" or "Need more info here" are not enlightening. Tell us what is right or what information is missing.)
- ◆ You'll notice occasional comments enclosed in <<< >>> brackets. These are notes about issues that need resolution or questions that need answering. Those that contain the word "Reviewer" or your name are questions for you to answer. Please do so.

What you're not looking for

Don't spend too much time proofreading. Mark things if you happen to see them, but don't dwell on grammar and editing. Technical review is for correcting technical material.

How to report comments and changes

If you're reviewing a .PDF:

- ◆ Record your comments in a Notepad or Word document.
- ◆ Refer to the topic title so we can find it in the book. (Page numbers are OK, but they change as the book changes.)

If you're reviewing a hard copy:

- ◆ Mark comments directly on the draft.
- ◆ Make sure your comments are legible.
- ◆ Use pen (pencil marks are harder to find). Colored pen is even better.

Delivering Beta Documentation

Frequently, we need to provide documentation to be included with the product beta. Beta documentation consists of the Release Notes and a .PDF version of the Reference Manual and Getting Started Guide. None of these documents are expected to be complete; we provide them as is.

This section describes some of the special steps you take to produce the following beta documentation:

- ◆ [Producing the Beta Release Notes](#)
- ◆ [Producing a Beta Manual](#) on page 148
- ◆ [Creating Beta Help](#) on page 149

Producing the Beta Release Notes

For the beta Release Notes, use the same format as for the final Release Notes, except add the following items:

1. Create a separate section for each new beta build. Add a line and link to each beta section in the Contents. Example: A beta Release Notes for WfWI 3.0 might contain links to the following sections:

Version 3.0 BETA #4 Bug Fixes/New Features/Enhancements
Version 3.0 BETA #3 Bug Fixes/New Features/Enhancements
Version 3.0 BETA #2 Bug Fixes/New Features/Enhancements
Version 3.0 BETA #1 Bug Fixes/New Features/Enhancements

2. Add the word BETA to the end of the first heading. Example:

Release Notes for Wise for Windows Installer 3.0 BETA.

3. Add this notice below the Introduction section:

IMPORTANT CONSIDERATION FOR BETA SOFTWARE

It is important to understand that the beta release of any product will not display the stability of a commercial Wise Solutions product. You might encounter problems with this release. This release is not appropriate for production use, and we strongly recommend that you back up your existing data before you install and run this software.

4. Add this verbiage to the Feedback section (verify the e-mail address with the testing department):

All feedback about this beta release, including comments, questions, and bugs, can be e-mailed to our testing department at beta@wise.com.

5. Delete the Wise Solutions Technical Support section from the bottom of the document.

Producing a Beta Manual

You can provide beta versions of the Reference Manual and Getting Started Guide, assuming you have made most of the documentation changes for the release. Beta documentation contains an additional chapter: Beta Notice. The Beta Notice simply contains the following text:

This document is a beta version of the <product name> <document name>. Portions of this manual are incomplete, and some elements, such as the Index, are missing.

To find information in this online manual, you can click on the Acrobat Reader bookmarks, click on the page numbers in the Table of Contents, or press Ctrl+F to search for key words.

To create a beta manual:

1. Copy the directory and subdirectories containing the Frame files and links to a new "Beta" directory. Do all the remaining steps in the Beta directory.
2. Copy the file Technical Publications\Doc Procedures&Templates\Beta Notice.fm into the Beta directory.
3. Open the file Beta Notice.fm and make the following changes:
 - Replace <product> with the product name such as Wise for Windows Installer.
 - Replace <document name> with the manual name, such as Reference Manual.
 - Make any other changes that are necessary for this particular beta release.
4. Open the book file and add the file Beta Notice.fm. Place it after the ToC and before the first chapter.
5. Follow the [Finishing Steps Checklist](#) on page 153, with these exceptions:
 - You do not have to delete all the comments in <<<>>> brackets. However, you should review them and edit or delete any that contain comments that a customer should not see.
 - You do not have to adjust the pagination.

Note

The Beta Notice will become chapter 1 and the remaining chapters will be renumbered. This doesn't matter for the beta version.

Creating Beta Help

<<<change this for HTML Help>>>

Because we create the online help from the finished documentation source files at the end of the documentation cycle, we cannot provide a beta help system. For most existing products, we simply ship the old help until the new one is available. For a new product, however, we must provide a "placeholder" help file so that beta testers do not receive an error when they press F1. The placeholder help consists of a single page containing the statement:

Help is not available for the beta version of <product name and version>. Please use the online Reference Manual instead.

The phrase *Reference Manual* is a hyperlink to the Reference Manual .PDF.

You can create a placeholder help file from scratch as needed, or you can use one of the standard placeholder help projects in the Technical Publications\Beta Help directory.

To use a standard placeholder help project:

1. If you don't know what the name of this product's help file should be, ask Development.
2. Create a new directory for the beta help project, such as WFWI 3.0 Beta Help.
3. Copy the contents of the appropriate Technical Publications\Beta Help directory to the new directory.
4. Open the help project and update the product name and version.

5. If necessary, replace the name of the online reference manual in the hyperlink.
6. Save and compile the help project, being sure to give it the correct name.
7. Deliver the .hlp and .cnt files to Development via SourceSafe.

Guidelines for ReadMes/Release Notes

What Not to Include

Do not include the following kinds of bugs in the Release Notes:

- ◆ Bugs that are found on new features during the development cycle, even if beta testers have found these bugs in beta builds. Reason: why should we expose our entire development cycle to all new users, reviewers, journalists, etc.
- ◆ Obscure bug found by our development team or QA team. Some bug that is not very common, that you have to do something very strange to get the bug, that no customer has ever complained about. Reason: why give users the impression that our product is very buggy if we don't have to.
- ◆ A bug that merely rectifies how something should have worked in the first place. Example: The focus in a list reverts to the top of the list every time you delete or add to the list. This is probably not something a user would have reported as a bug in the first place, and is really just improving the UI in an unobtrusive way. Reason: adding this type of thing to release notes 1) makes the bug list look unnecessarily long; 2) users probably didn't consciously notice it anyway; 3) it just made something work the way it should have in the first place.
- ◆ Bugs that were fixed with coding. Example: There is a bug that says "when users add a directory name that starts with a period, problems occur." The fix for this bug is to display an error message if the user tries to name a directory starting with a period. There is no need to document this, because it is self-documenting. Only people who try to do this will need the doc, and only they will get it, in the form of an error message. There is no need to doc it in the Release Notes because 1) it is trivial; 2) you have to do an odd thing to experience the problem; 3) anyone who tries to do this odd thing will immediately be warned of the limitation.

Documenting New Features

When listing a new feature in the Release Notes, include enough information for people to use it.

Examples:

- 1) "A new Download Merge Module wizard gives you the ability to link to Wise Solutions or to other developers to download the latest merge modules."
- 2) "A new Download Merge modules wizard gives you the ability to link to Wise Solutions or other developers to download the latest merge modules. To access the new wizard, select help menu > Download Merge Modules."

Example 1 tells about the new feature, but puts the onus of finding and using the feature on the user. Because we don't provide beta doc, it's more important to give the user enough information to at least get started with a new feature, because this is the source of our beta testing. If a new feature seems straightforward, but has some quirky weird thing you have to do to get it to work, then we should tell them that in the Release Notes, just to enable testing of the new feature.

Including Links to Previous Release Notes

Each Release Notes document covers a single point release (example: 4.02) and can contain hyperlinks to Release Notes for previous versions of the major release (4.01 and 4.0). The most current Release Notes document is always named <product>.htm, and the Release Notes for older versions have the version number appended to the file names. Example: If the current version of WfWI is 4.2, then the 4.2 Release Notes

document is named WfWI.htm. Release notes for previous versions of WfWI are named WfWI402.htm, WfWI401.htm, and WfWI40.htm.

In the "Contents" section of the Release Notes, add hyperlinks to the previous versions. Example: The Contents section of the WfWI 4.2 Release Notes might look like this:

Enhancements in 4.2
 Other Changes in 4.2
 Wise for Windows Installer 4.02 Release Notes
 Wise for Windows Installer 4.01 Release Notes
 Wise for Windows Installer 4.0 Release Notes

When we do release 5.0, however, we will start over with a new set of Release Notes; we will not refer to any of the 4.x versions.

Creating Release Notes for a New Version

When you start working on a new version of a product, set up the new Release Notes documents.

1. Check out the current Release Notes document (<product>.htm) from SourceSafe.
2. Make a copy of the <product>.htm and append the previous version number to the file name. Example: If the file you checked out was for WfWI 4.2, then name the copied file *WfWI42.htm*.
3. Open the <product>.htm file and do the following:
 - Delete the items that applied to the previous version, You can leave the tables that form the various sections.
 - Change the version number wherever applicable, such as in the title at the top of the page and in the various sections.
 - In the Contents section, add lines for hyperlinks to the new sections for this release. Remember to change the <A HREF> tag in each section.
 - In the Contents section, add the appropriate file names to the hyperlinks that point to the older files. Example: If the line *Enhancements in 4.2* links to *#Enhancements42*, you would change the link to *WfWI42.htm#Enhancements42*.
 - Add items that apply to the new version.
4. Save the file.
5. Add the new file (example: WfWI42.htm) to SourceSafe and notify Development to add it to the installation.

Finishing Steps Checklist

To prepare a book for conversion to .PDF, PostScript, and online help, complete these steps. Print the checklist and use it to keep track of the steps as you complete them.

If you make any changes to the book after completing these finishing steps, you must repeat the steps.

Note

Complete ALL of the following steps if you are making a .PDF. If you are skipping the .PDF and starting help, just do the steps marked with an asterisk.

X	Step	Notes
	Remove all obsolete files and directories from the book folder.	If any files remain linked to the deleted files, it will result in broken x-refs and image refs, which you can fix during Finishing Steps.
	Open the book file (.book) in Frame.	Note: WfWI and Windows Installer Editor have separate book files.
	Show all conditional text and show the condition indicators.	
	Check and update the book's File Info (properties).	Select all files, then select File > File Info. See Check and Update the Book's Properties on page 155.
	Make sure the chapters are in the correct order in the book.	Use an archived .PDF as a guide if necessary.
	Reset the document numbering.	Select all files in the book. (Format > Document > Numbering.) See Reset the Document Numbering on page 155.
	Check and update the PDF Setup.	Select all files in the book. (Format > Document > PDF Setup.) See PDF Settings on page 169.
	* In the Front Page file, check and update the copyright date and version number. Also update the copyright date in the HelpFooter marker in Welcome.	See Update the Copyright Date and Version Number on page 156.
	* Check the contents of the HelpHeader marker in the Welcome chapter (also in chapters that become stand-alone help).	<ul style="list-style-type: none"> ◆ For beta help, add BETA to the end of the marker text. ◆ For final help, make sure the BETA has been removed.
	* Find all <<< and >>> and resolve any remaining issues. If making beta help, just apply the comment condition to all <<<>>> text.	
	SAVE ALL OPEN FILES	Doing entire book searches often crashes Frame.

X	Step	Notes
	* Find and fix unresolved cross-references.	If the book involves copying and pasting from one book to another, make sure cross-references go to the correct book. See Search For and Fix Unresolved Cross-References on page 157.
	SAVE ALL OPEN FILES	Doing entire book searches often crashes Frame.
	* Find and fix unresolved text insets.	Caution: When fixing text insets, make sure that condition tags in the target file are not removed.
	* Find and fix conditional inset problems	Search for "conditional inset" and make sure the proper style and conditional text tags are applied.
	Close the book. Save a copy of the book with change bars in a separate directory for the Curriculum Developer. Re-open the book.	
	Turn off and clear change bars.	Select all files in the book.
	Show and hide conditional text.	Select all files in the book. Most books have condition templates that facilitate this step. See Changing the Book View on page 158. Otherwise, see Hide Conditional Text on page 157.
	Re-check for unresolved cross-references.	
	* Do a final spell check.	
	* Update the book. See Update the Book on page 160.	If the Inconsistent Numbering Properties dialog appears, cancel the operation and reset the numbering. See Reset the Document Numbering on page 155.
	Edit the TOC.	See Edit the Table of Contents on page 160.
	Format the Index (optional).	See Format the Index on page 160.
	* Save the book.	
	Make the product PDF.	Change the Distiller settings and the default printer. See Creating a .PDF File on page 167.
	Open the .PDF and page through it.	Look for: ◆ TOC problems ◆ Chapter numbering problems ◆ Extra blank pages ◆ Bad formatting due to conditional text errors
	Set .PDF options and replace the cover page of the finished .PDF. Save the .PDF.	See Creating a .PDF File on page 167.
	If this book will NOT be printed, skip to the next-to-last step.	
	(GSGs only.) Convert the book to 5.5x8.5 format.	Apply the 5x8 template and update the book. See Converting to 5x8 Format on page 160.
	(GSGs only.) Check pagination.	See Check Pagination on page 159.
	(WfWI & Wise only.) Convert the book to 7x9 format.	Apply the 7x9 template, check callout frames. Update the book. See Converting to 7x9 Format on page 161.

X	Step	Notes
	Make the PostScript file (if this is the final build).	Change the Distiller settings and the default printer. See Saving the Book as PostScript on page 172
	Make the quick-print .PDF (if this is the final build).	See Creating a .PDF File for Quick Printing on page 177.
	(GSGs, WfWI, & Wise only.) Convert the book back to 8.5x11 format.	Apply the 8.5x11 template. Remove pagination overrides. See Converting to 5x8 Format on page 160 or Converting to 7x9 Format on page 161.
	Reset the conditions to Show All (mark Show Condition Indicators). Save and close the book.	Select all files in the book.
	Send final PDFs and Release Notes to Lindon office so they can post them on the Latest Documentation page of the Altiris Web site.	Send them to Rajeev Soorma and cc Jon Sharp. Instruct them to post them so that they appear on the Altiris Web site the day of the release.

Check and Update the Book's Properties

The Frame book has properties that appear in the resulting PDF. If you've created a new book, you must set these properties. If you've updated an existing book, you must check the properties and change them if necessary (example: if the product name has changed).

1. Open the book file (.book) in FrameMaker.
2. Select all files in the book.
3. Select File > File Info.
4. In **Author**, enter Wise Solutions, Inc.
5. In **Title**, enter the book's title, for example Wise Package Studio Reference.

Note

For the Windows Installer books, just enter Reference Manual, because the title is different for each book.

6. Click the Set button.

Also see [Finishing Steps Checklist](#) on page 153.

Reset the Document Numbering

1. Open the book and select all the chapters.
2. Select Format > Document > Numbering and set the numbering as follows:
 - Volume tab: Mark **Volume #** and enter 1
 - Chapter tab: Mark **Continue Numbering from Previous File in Book**
 - Page tab: Mark **Continue Numbering from Previous Page in Book**
 - Paragraph tab: Mark **Continue Numbering from Previous Paragraph in Book**
 - Footnote tab: Mark **Start Over on Each Page**
 - Table Footnote tab: Make sure **alphabetic (n)** is selected

3. Click Set.
4. Update the book.

Open each chapter in the book to check the chapter number. If a chapter is numbered 1 when it shouldn't be, see [Chapter is Numbered 1](#) on page 226.

Also see [Finishing Steps Checklist](#) on page 153.

Check and Update the Book's PDF Setup

With Frame 7.0, you can save .PDF settings in the book and chapter files. If you've created a new book, you must set these settings. If you've updated an existing book, you must check the settings and change them if necessary.

Note

Per Adobe, make sure the settings are the same for all files in the book as well as for the book file. Example: If one chapter file has different bookmark settings, the .PDF bookmarks will not be made correctly.

1. Open the book file (.book) in FrameMaker.
2. Select ALL files in the book.
3. Select Format > Document > PDF Setup.
The PDF Setup for Selected Files dialog appears.
4. Set the PDF settings as described in [PDF Settings](#) on page 169. When you finish, click the Set button.

Also see [Finishing Steps Checklist](#) on page 153.

Update the Copyright Date and Version Number

Page 2 of every Wise Solutions book contains a copyright notice, trademark information, and the product version number. As of 2003, the books' covers no longer display the version number. The version number on page 2 lets you see at a glance which actual version of the book you have.

Guidelines for Specifying the Copyright Year

- ◆ When you first publish a document, specify the year of first publication.
- ◆ When you update a document, append the current year, if it is different from the first year of publication.

Example: The first time you update a book, the copyright notice might look like this:

©1999, 2000 Wise Solutions, Inc.

During subsequent updates, you omit the intervening copyright dates and specify only the first and last dates, like so:

©1999-2002 Wise Solutions, Inc.

- ◆ If you rework a document such that more than 50% of its content is different from previous version, you can start over with a single copyright year. This is a judgement call; when in doubt, retain the original copyright year.

To update the copyright date and version number:

1. Open the book's Front Page.fm and go to page 2.

2. In the first heading at the top of the page, update the version number.
3. Under the heading *Copyright*, update the copyright date.
4. Check with Wise's corporate counsel to see if we have any new trademarks. If so, add them under the *Trademarks* heading. Arrange them in alphabetical order.

Note

If you add new trademarks, open the document Front Page.fm in Technical Publications\Admin\DocProcedures&Templates\Current Doc Procedures and change them there as well.

5. Save and close the file.
For Getting Started Guides, which are not converted to PDF, skip the steps below that pertain to the HelpFooter.
6. Open the book's Welcome chapter.
7. Find the HelpFooter marker (it should be in the first sentence on the first Welcome page).
Edit the marker so that it contains the new copyright date. (Select the marker and then select Special > Marker. Remember to click the Edit Marker button when you finish.)
8. If the book contains a chapter that becomes a stand-alone help file, you must update the HelpFooter in that chapter as well. Examples are the Debugger chapter in WfWI and WIE, and the Test Expert chapter in Workbench.

Also see [Finishing Steps Checklist](#) on page 153.

Search For and Fix Unresolved Cross-References

Sometimes, if you copy and paste between books, cross-references can inadvertently stay linked to the book from which text was copied. (Avoid this by not copying cross-references.) To check for this situation, follow the procedure below.

Before you search for unresolved cross-references:

1. Clean up the book directory, putting all unused Frame files in a directory named "unused". That way, all cross-refs to obsolete files will break immediately.
2. Change the names of all current book directories. Example: If you are currently working on the books for WfWI 4 and WIE 3.1, change both the top level directory names. This causes any cross-refs to the other book's directory to break immediately.

After you've performed the steps above, search for unresolved cross-references as you normally do. You can use Edit > Update References to link cross-refs to different files.

Also see [Finishing Steps Checklist](#) on page 153.

Hide Conditional Text

We use conditional text for internal comments, online help navigation, and for product-specific information in single sourced documents. Depending on which book you're working with, you must hide some of the conditional text before creating the .PDF and PostScript files.

We use special templates to change the condition settings in single-sourced books. For details, see [Changing the Book View](#) on page 158.

To change conditional text settings for the entire book without using a template, open the book file, select all its files, then choose View > Show/Hide Conditional Text. The table below lists the condition tags to show for each book. Hide all tags that are not listed.

Book	Show	Show Condition Indicators
Any book other than the ones listed below	not for help	clear
Getting Started for Package Studio	not for help WPS	clear
Getting Started for Quality Assurance module	not for help QA	clear
Software Manager for Package Studio	not for help WPS	clear
Software Manager Ref for WfWI	not for help WfWI only	clear
Windows Installer Editor Reference	not for help WIE only	clear
Wise for Windows Installer Reference	not for help WfWI only	clear
Wise Installation System Reference	not for help WIS WIS+WSE	clear
WiseScript Editor	not for help WIS+WSE WSE	clear
WiseScript Express	Express not for help	clear

Also see [Finishing Steps Checklist](#) on page 153.

Changing the Book View

To change a single-sourced book to a view that represents a specific version, you apply a condition template. Condition templates contain conditional text settings and variable definitions that are specific to one version of a single-sourced book. You apply the condition template to a book to:

- ◆ Print the book or portions of the book for review.
- ◆ Check for unresolved cross-references.
- ◆ Create a .PDF of the book.
- ◆ Create a PostScript of the book.
- ◆ Archive the book at the completion of a release,

At all other times, such as when you are working on the book's contents, it is best to show all conditions and show condition indicators.

To change the book view:

1. In Frame, open the appropriate condition template.
2. Open the book file and select all files in the book.
3. Select File menu > Import > Formats.
4. In the Import Formats dialog:
 - From **Import from Document**, select the condition template file.
 - Deselect all options.
 - Mark these checkboxes:
 - **Variable Definitions**
 - **Cross-Reference Formats**
 - Click the Import button.

The book view is changed to a specific version.

Also see:

[Finishing Steps Checklist](#) on page 153

[Creating Condition Templates](#) on page 131

Check Pagination

For most books, we do not adjust pagination because it's too difficult to manage when using conditional text and applying different formats for some books.

We do need to check and adjust pagination when switching between large and small formats. This is especially a problem when switching to the 5x8 format for Getting Started Guides.

To adjust pagination:

1. Open one chapter at a time and page through it to see if you need to adjust its pagination. Use your judgement when deciding how to break a page.
2. To fix pagination, select Special > Page Break and mark **At Top of Next Available Page**. You also can use the "Keep With: Next Pgf" or "Start: Top of Page" paragraph attributes (do not Update All).

To remove pagination overrides:

This requires the Cudspan Doc Tools plug-in. See [Tools for Creating Wise Documentation](#) on page 69.

1. Select all files in the book.
2. Select CudSpan Book Tools > Booklooker > Hunt Pgf Overrides for Current Book.
A new Frame document opens, listing hotlinks to the paragraphs where overrides occur.
3. Click each link in turn and remove the override.
4. Save the book.

Also see [Finishing Steps Checklist](#) on page 153.

Update the Book

1. Open the book file (.book) in FrameMaker.
2. Select Edit > Update Book.
The Update Book dialog appears.
3. Verify that the TOC file name and the index file name appear in the Generate list, and click Update.
At this point you might see a Book Error Log window. If this happens, fix the problems listed (example: unresolved cross-references) and update the book again.
4. If the Inconsistencies in Book dialog appears, mark **Skip Remaining Consistency Messages** and click Continue.

Also see [Finishing Steps Checklist](#) on page 153.

Edit the Table of Contents

1. In FrameMaker, open the TOC file.
2. Look through the TOC for headings that flow onto a second line. If the line breaks in a logical place and in an attractive way, you can leave it alone. If the line breaks awkwardly, go to that part of the book to see if you need to force a more attractive line break. For guidelines, see [Forcing a Line Break in Headings](#) on page 87.
3. (Optional.) Paginate the TOC so the pages break at appropriate spots. Eliminate widow and orphan TOC entries. You also might have to fiddle with the pagination because deleting returns in middle of headings usually makes the TOC shorter, and might throw off the page count. Make sure the TOC ends up with the same number of pages it started with.

Also see [Finishing Steps Checklist](#) on page 153.

Format the Index

This step is optional; it's nice to do if you have time, but it's not necessary. Page through the index and look for widowed and orphaned topics and subtopics. Example: If a topic is by itself at the bottom of a column, and all its subtopics are at the top of the next column, it's nice to keep the topic with its subtopics. This is more important when the topic is on one page and its subtopics are on another. It's OK to break a topic in the middle of a list of subtopics.

To keep a topic with its subtopic, apply the paragraph attribute Keep With: Next Pgf.

Also see [Finishing Steps Checklist](#) on page 153.

Converting to 5x8 Format

As of June 2004, we use the 5x8 format for Getting Started Guides that are commercially printed. This format (actually 5.5x8.5) fits the Altiris-style packaging. It also is paginated for double-sided printing, and forces each chapter to have an even number of pages, which is desired for printed books.

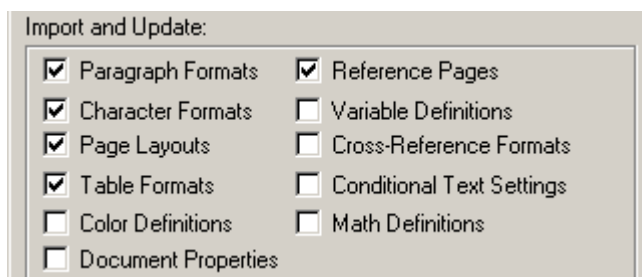
The default template for these books is 8.5x11. We change the format to 5x8 before preparing the files for printing, and then change the format back to 8.5x11.

Note

Do not use this procedure to convert a book to 7x9 format. See [Converting to 7x9 Format](#) on page 161.

To convert a book to 5x8 format:

1. In Frame, open the 5x8 Template.fm file.
2. Open the book file and select all files in the book *except* the index.
3. Select File menu > Import > Formats.
4. On the Import Formats dialog:
 - From **Import from Document**, select **5x8 Template.fm**.
 - Deselect all options.
 - Mark these checkboxes and click Import:



5. Open the 5x8 IX Template.fm file.
6. In the book, select the index file only.
7. Select File menu > Import > Formats.
8. On the Import Formats dialog:
 - From **Import from Document**, select **5x8 IX Template.fm**.
 - Verify that the options you selected above are still selected and click Import.

The fonts and page layouts are optimized for the 5x8 format. An extra page is added to each chapter if necessary.
9. Page through the book and look for anchored frames and tables that need to be resized. Because the main text frame is much smaller than in the other formats, it is possible that some anchored frames and tables might be too large. See [5x8 Format Considerations](#) on page 77.
10. Update the book.

Now you can prepare the file for printing. See [Creating a PostScript File for Commercial Printing](#) on page 171 or [Creating a .PDF File for Quick Printing](#) on page 177.

When you finish preparing the file for printing, convert the book back to 8.5x11 format. See [Converting to 8.5x11 Format](#) on page 162.

Also see [Finishing Steps Checklist](#) on page 153.

Converting to 7x9 Format

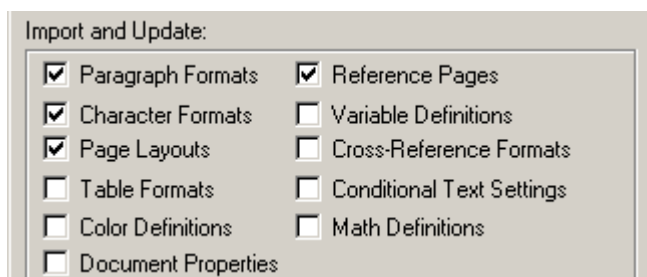
As of March 2004, we use the 7x9 format only for books that are commercially printed, other than Getting Started Guides. Currently, this includes WfWI and Wise Installation System references only.

The default template for these books is 8.5x11. We change the format to 7x9 before preparing the files for printing, and then change the format back to 8.5x11.

Note

Do not use this procedure to convert a book to 5x8 format. See [Converting to 5x8 Format](#) on page 160.

1. In Frame, open the 7x9 Template.fm file.
2. Open the book file and select all files in the book *except* the index.
3. Select File menu > Import > Formats.
4. On the Import Formats dialog:
 - From **Import from Document**, select **7x9 Template.fm**.
 - Deselect all options.
 - Mark these checkboxes and click Import:



- Click Import.
5. Open the 7x9 IX Template.fm file.
 6. In the book, select the index file only.
 7. Select File menu > Import > Formats.
 8. On the Import Formats dialog:
 - From **Import from Document**, select **7x9 IX Template.fm**.
 - Verify that the options you selected above are still selected and click Import.

The fonts and page layouts are optimized for the 7x9 format. An extra page is added to each chapter if necessary.
 9. Page through the book and look for callout frames that need to be resized. Because the fonts in the 7x9 template are slightly larger, it is possible that some callout frames might be too small. (Tip: Search for anchored frames.)
 10. Update the book.

Now you can prepare the file for printing. See [Creating a PostScript File for Commercial Printing](#) on page 171 or [Creating a .PDF File for Quick Printing](#) on page 177.

When you finish preparing the file for printing, convert the book back to 8.5x11 format. See [Converting to 8.5x11 Format](#) on page 162.

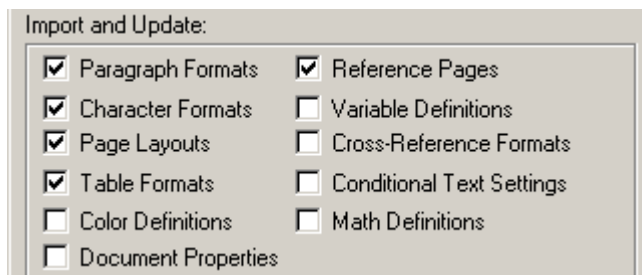
Also see [Finishing Steps Checklist](#) on page 153.

Converting to 8.5x11 Format

After you prepare a book for printing, convert it back to 8.5x11 format. This is the default format for all books, and is optimized for producing a .PDF. It has single-sided

pagination, and removes any blank pages at the end of chapters, because .PDFs don't need to have an even number of pages.

1. In Frame, open the 8.5x11 Template.fm file.
2. Open the book file and select all files in the book *except* the index.
3. Select File menu > Import > Formats.
4. On the Import Formats dialog:
 - From the **Import from Document**, select **8.5x11 Template.fm**.
 - Deselect all options.
 - Mark these checkboxes and click Import:



5. Open the 8.5x11 IX Template.fm file.
6. In the book, select the index file only.
7. Select File menu > Import > Formats.
8. On the Import Formats dialog:
 - From **Import from Document**, select **8.5x11 IX Template.fm**.
 - Verify that the options you selected above are still selected and click Import:
The fonts and page layouts are optimized for the 8.5x11 format. Blank pages are removed from the end of chapters.
9. Save the book.

Also see [Finishing Steps Checklist](#) on page 153.

Housekeeping

After you've delivered the project, perform these housekeeping tasks:

- ◆ Clean out or delete temporary files and directories.
- ◆ Clean out the FrameMaker graphics directories, if you didn't have time to do it while working on the project. See [Cleaning Up Graphics Directories](#).
- ◆ Archive the project files; see [Archiving Documentation Files](#) on page 124.
- ◆ Update the Doc Procedures book if you have changed any procedures.

Cleaning Up Graphics Directories

Open the book in Frame. For each chapter, do the following:

1. Open the chapter.
2. Select Special > Index of > References to generate an index of references to imported graphics.
3. Are there any references to graphics in a different project directory?

Hint: Look for this kind of directory path.

```

.....
.../ConflictMgr/ConflictMgr Ref/Basicstifs/INIFileIcon.tif @ 50 dpi 183
.../ConflictMgr/ConflictMgr Ref/Basicstifs/PathIcon.tif @ 50 dpi 183
.../ConflictMgr/ConflictMgr Ref/Basicstifs/ShortcutIcon.tif @ 50 dpi 183
TemplateTifs/PDFSetupBookmarks.tif @ 115 dpi 173
TemplateTifs/PSPageSize.tif @ 96 dpi 182
.....

```

If so, copy them to the correct directory and re-import to the Frame file.

4. Print the index of references.
5. Go to Windows Explorer and look at the graphics directory for the chapter you're working on. Delete all unused files. If in doubt, ask.

Chapter 11

Creating .PDF and PostScript Files

We provide .PDF manuals with our products. We also need to create .PDF files for quick printing, and PostScript files for commercial offset printing.

Topics include:

- ◆ [Setting Up Acrobat Distiller.](#)
- ◆ [Creating a .PDF File.](#)
- ◆ [Creating a PostScript File for Commercial Printing.](#)
- ◆ [Creating a .PDF File for Quick Printing.](#)

Setting Up Acrobat Distiller

Note

The following procedures assume you are using Adobe Acrobat Distiller version 6.0.

The easiest way to set up your Acrobat Distiller for creating Wise .PDFs is to copy the existing job options file.

If you will be preparing files for printing at West Wind Litho (Altiris' printing supplier), you also need to copy their specific job options.

To copy Wise job options to your Acrobat Distiller:

Copy the file WiseDoc.joboptions

From: Technical Publications\Admin\Documentation Software\Acrobat 6

To:

- ◆ Documents and Settings\All Users\Documents\Adobe PDF 6.0\Settings
This is where Acrobat 6.0 looks for it.
- ◆ Program Files\Adobe\Acrobat 6.0\Distillr\Settings
This is where Frame looks for it.

To set up Adobe Acrobat Distiller for making Wise .PDFs:

<<<need to update this for Acrobat 6.0 - JK 8/20/04>>>

This is what you have to do to set your Distiller options from scratch. *You do not have to do this* if you have copied the WiseDoc.joboptions file.

1. Launch Acrobat Distiller.
2. From the **Job Options** drop-down list, select **Print**.
3. Select Settings > Job Options.
4. On the General tab, select **Acrobat 3.0** in the **Compatibility** drop-down list.
5. Clear the **Optimize for Fast Web View** checkbox.
6. Click Save As and save these options with the name *WiseDoc*. You can use these saved job options to create .PDFs in the future.
7. Close the Job Options dialog.

** Do NOT set any security.

Next, in FrameMaker, you save the book as a .PDF.

To copy job options for printing at West Wind Litho:

Copy the file Win-Acr4-Preps.joboptions

From: Technical Publications\Admin\Documentation Software\Acrobat 6

To:

- ◆ Documents and Settings\All Users\Documents\Adobe PDF 6.0\Settings
This is where Acrobat 6.0 looks for it.
- ◆ Program Files\Adobe\Acrobat 6.0\Distillr\Settings
This is where Frame looks for it.

Creating a .PDF File

Note

The following procedure assumes you are using Adobe Acrobat Distiller version 6.0.

If you have set up Acrobat Distiller, you can save the book as a .PDF. Although there are multiple ways to create a .PDF, we use the following steps because the resulting .PDF is nearly 50% smaller than if we simply print to PostScript in Frame and then distill that PostScript.

Before you begin, verify that you have a .PDF of the cover page from Wise's graphic design department. See [Deliverables from Wise's Graphic Design](#) on page 144.

1. Open the book file (.book) in FrameMaker.
2. Select File > Print Setup.
3. Select **Adobe PDF** from the **Name** drop-down list and click OK.
4. Select File > Save Book As, change the **File name** extension to **.pdf**, change **Save as type** to **PDF**, and click Save.

The PDF Setup for Selected Files dialog appears.

5. Set the PDF settings as described in [PDF Settings](#) on page 169. When you finish, click the Set button in the PDF Setup dialog.

FrameMaker creates a temporary PostScript file (.TPS) from which Distiller will create the .PDF. As soon as the .TPS file is created, Distiller opens and begins creating the .PDF.

After a few moments, the .PDF appears in the directory you specified in the Save As... dialog. It is not yet finished, because the first page of the book is a placeholder page.

6. Open the resulting .PDF in Acrobat.
7. Select File > Document Properties. In the list on the left of the Document Properties dialog, click Initial View.

The Document Open Options dialog appears.

8. Set the options exactly as shown below and click OK.

The image shows three panels of settings for a PDF viewer:

- Document Options:**
 - Show: Bookmarks Panel and Page (dropdown)
 - Page layout: Default (dropdown)
 - Magnification: Default (dropdown)
 - Open to: Page number: 1 of 59
 - Last-viewed page
- Window Options:**
 - Resize window to initial page
 - Center window on screen
 - Open in Full Screen mode
 - Show: File Name (dropdown)
- User Interface Options:**
 - Hide menu bar
 - Hide tool bars
 - Hide window controls

9. Select Document > Pages > Replace.
The Select File With New Pages dialog appears.
10. Open the cover page .PDF you obtained from the graphic design department. Make sure you're in the right directory—it's easy to pick the wrong cover. In the Replace Pages dialog, choose to replace page 1 to 1 of the book .PDF with page 1 of the cover page .PDF.

Note

The cover for single sourced books differs depending on the product you are shipping it with. Make sure you use the correct cover.

11. Collapse all the TOC Bookmarks: at the top of the Bookmarks pane, click Options and select Collapse Top-Level Bookmarks.
(Currently .PDFs are created with bookmarks expanded. This makes the TOC less useful, especially in a large book like WfWI. The setting that is supposed to control this is not working at present so we have to collapse the TOC manually.)
12. Select File > Reduce File Size.
This does a Save As and dramatically reduces the final size of the .PDF by as much as 35% (the larger the book, the greater the reduction).

Also see [Finishing Steps Checklist](#) on page 153.

PDF Settings

With Frame 7.0, you can save PDF settings in the book and chapter files. If you've created a new book, you must set these settings. If you've updated an existing book, you must check the settings and change them if necessary.

IMPORTANT

Per Adobe, make sure the settings are the same for all files in the book as well as for the book file. Example: If one chapter file has different bookmark settings, the PDF bookmarks will not be made correctly.

You set the PDF settings in the PDF Setup dialog. You access this dialog in several ways, depending on the stage of production you are in.

- ◆ If you are following the Finishing Steps Checklist, you check the settings for ALL FILES in the book. Select all files, then select Format > Document > PDF Setup.
- ◆ If you are making a .PDF, the PDF Setup dialog appears when you select the command to save the book as a .PDF.
- ◆ If you are making a PostScript file, the PDF Setup dialog appears when you click the PDF Setup button in the Print Book dialog.

In the PDF Setup dialog, make sure the settings are as follows.

Settings tab:

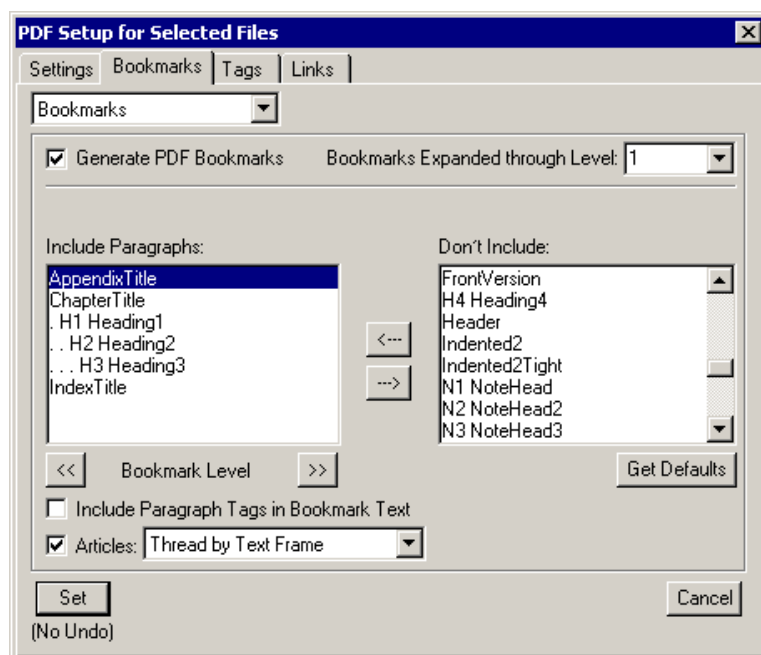
In the **PDF Job Options** drop-down list,

- ◆ If you are following the finishing steps or making a product .PDF, select **WiseDoc**.
- ◆ If you are making a PostScript file for commercial offset printing, select **Press**.
- ◆ If you are making a PostScript/.PDF file for quick printing, select either **Press** or **Win-Acr4-Preps**. See [Creating a .PDF File for Quick Printing](#) on page 177.

Leave the defaults in the rest of the fields.

Bookmarks tab:

Set the options exactly as shown in the following screen shot.



Note

We would prefer to set the **Bookmarks Expanded through Level** setting to **None**, but due to an Adobe bug, that doesn't work. See Adobe knowledgebase article 318572, or call Adobe Support and refer to our support case number 5419391. JK 01/03

Tags tab:

Clear the **Generate Tagged PDF** checkbox; this reduces the size of the resulting .PDF file.

Links tab:

Clear the **Create Named Destinations for All Paragraphs** checkbox.

Also see [Finishing Steps Checklist](#) on page 153.

Creating a PostScript File for Commercial Printing

When we have a manual printed by a commercial offset printer, we typically provide a PostScript file. To create a PostScript file, you must set up a PostScript printer driver if you have not already done so, then save the FrameMaker book as PostScript.

Note

Occasionally, Customer Service runs out of manuals before a new order arrives. In such cases, they have copies made at a quick printer. The requirements for creating a file for quick printing are a bit different than for high-quality printing. See [Creating a .PDF File for Quick Printing](#) on page 177.

Also see:

[Deliverables to the Printer](#)
[Installing the Generic PostScript Printer Driver and Setting It Up](#)
[Saving the Book as PostScript](#)
[Testing the PostScript File](#)
[Printing a Hard Copy for Proofing](#)
[Sending the Book to the Printer](#)
[Reviewing the Printer's Proof](#)

Deliverables to the Printer

On an initial order, you provide the following items to the printer:

- ◆ PostScript file of the book.
- ◆ Hard copy of the book, printed at 82% with registration marks.
- ◆ Art for cover and title page, if Graphic Design has not given it to the printer already.

When you order a book reprint, you only have to provide a new file and hard copy IF the contents of the book have changed. (Example: If we had two maintenance releases since the initial book printing, you should provide those changes to the printer.)

Sometimes, if the changes are minor, you can provide just the changed pages. It costs less because the printer doesn't have to redo the plates for the entire book. However, if the changes are extensive, or if they cause pagination changes or, even trickier to find, cross-reference changes, it's better to provide a complete new PostScript file. Use your judgement.

The requirements discussed here might differ slightly with each printing company you use. Before delivering the PostScript file, check with the printer you will use to verify their requirements and perform a test.

Also see [Creating a PostScript File for Commercial Printing](#) on page 171.

Installing the Generic PostScript Printer Driver and Setting It Up

Note

As of Oct. 2000, we use Northwest Graphics to print our manuals. The procedures below have been tested and verified by Northwest. If you use a different printing company, you should send them a test file to make sure they can output it. If possible, do this *before* you are in a deadline situation!

To install the "Generic PostScript Printer":

1. Install Adobe's PostScript Drivers. You can obtain this installation file either in the Tech Pubs Documentation Software folder, or from Adobe's Web site.
2. During the installation, mark the **It is directly connected to your computer (Local Printer)** option.
3. On the Local Port Selection dialog, select **FILE: LOCAL PORT**.
4. On the Select Printer Model dialog, select Generic PostScript Printer.
5. Proceed through the rest of the installation.

To set up a PostScript driver:

Make sure the driver properties are set as follows:

1. From the Windows Start menu, select Settings > Printers.
2. Right-click the Generic PostScript Printer icon, then select Properties.
3. On the General tab, click the Printing Preferences button.
4. On the Printing Preferences dialog, click the Layout tab, then click the Advanced button.
The Advanced Options dialog appears.
5. Expand the **Graphic** tree.
6. Click the link to the right of **TrueType Font** and select **Download as Softfont** from the drop-down list that appears.
7. Expand the PostScript Options tree under the Document Options.
8. Click the link to the right of **TrueType Font Download Option** and select **Outline** from the drop-down list that appears.
9. Leave the default settings for all other options.
10. Click OK to exit the Advanced Options dialog, then continue to click OK to close the Printing Preferences and Properties dialogs.

Also see [Creating a PostScript File for Commercial Printing](#) on page 171.

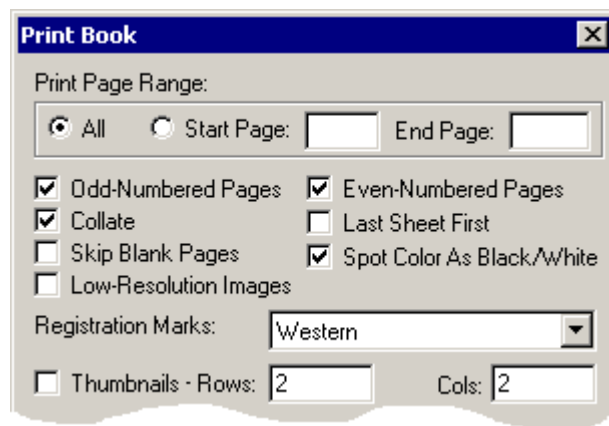
Saving the Book as PostScript

In most cases, PostScript is the best format to provide to a commercial printer. Before doing this, set up the PostScript Driver up as described in [Installing the Generic PostScript Printer Driver and Setting It Up](#) on page 172.

Remember that the first page of the Frame book is a placeholder page, and must be replaced by the printer with the title page art provided by the graphic design department. See [Deliverables from Wise's Graphic Design](#) on page 144.

Also see [Finishing Steps Checklist](#) on page 153.

1. Open the book file (.book) in FrameMaker.
2. If necessary:
 - Apply the appropriate condition template.
 - Apply the appropriate format (5x8 or 7x9).
3. Select File > Print Setup. Select **Generic PostScript Printer** from the **Name** drop-down list and click OK.
4. Select File > Print Book.
The Print Book dialog appears.
5. Complete the top part of the dialog:



6. Complete the remainder of the Print Book dialog:
 - **Scale**
 - For 5x8 format, enter 100%.
 - For 7x9 format, enter 82%.
 - **Print to File**
Mark this checkbox, and then enter a file name that ends with the extension .ps. Set the path to the PDFs_PostScripts/product PDF directory under the book directory. Single-sourced books typically have an additional directory level.
 - **Generate Acrobat Data**
Mark this checkbox and then click the PDF Setup button.
The PDF Setup dialog appears.
7. Click the Settings tab and select **Press** from the **PDF Job Options** drop-down list.
8. Set the remaining PDF settings as described in [PDF Settings](#) on page 169. When you finish, click the Set button in the PDF Setup dialog.
Wait while the book is updated; this might take a few minutes.
9. In the Print Book dialog, click the Print button.
The Printing Book dialog appears as the PostScript file is created.
The finished PostScript file (.PS) appears in the directory specified in the Print Book dialog.
10. If you are delivering this PostScript file to a printer, see [Testing the PostScript File](#) on page 174.

11. If you are creating a PDF for quick printing, now see [Creating a .PDF File for Quick Printing](#) on page 177, which directs you to distill the PostScript file and test the resulting .PDF.

Also see [Creating a PostScript File for Commercial Printing](#) on page 171.

Testing the PostScript File

When you create a PostScript file, you should test it before sending it to a commercial printer. You test the PostScript file by distilling it to a .PDF and then viewing the .PDF. This lets you check the fonts and the pagination. If you have a .PDF file in the same folder as the PostScript with the same name, this test will fail.

If you are also creating a .PDF for quick printing, skip this procedure and go directly to [Creating a .PDF File for Quick Printing](#) on page 177, which directs you to distill the PostScript file and test the resulting PDF.

1. Make sure the PostScript file (.PS) and the product .PDF file are in different directories; otherwise, the .PDF will get overwritten during this procedure.
2. In Windows Explorer, double-click the .PS file. Distiller opens and distills the file to a .PDF
3. Open the resulting .PDF in Acrobat and check the following:
 - Make sure the correct fonts are used and are smooth instead of jagged.
 - Spot-check pages throughout the book and compare them to the hard copy (see [Printing a Hard Copy for Proofing](#)). Differences in the pagination could be caused by a font problem.
4. When you finish checking the file, close the .PDF.

Also see [Creating a PostScript File for Commercial Printing](#) on page 171.

Printing a Hard Copy for Proofing

When you first place an order to print a book, you must provide a hard copy to the printer to use as a reference. Typically, you do not need to print a hard copy for reorders.

1. Open the book file (.book) in FrameMaker.
2. Select File > Print Setup, select your usual printer and click OK.
3. Select File > Print Book.
4. The Print Book dialog appears.
5. Complete the dialog:
 - **Print Page Range**
Select **All**.
 - Mark the following checkboxes:
Odd-Numbered Pages
Collate
Even-Numbered Pages
Spot Color As Black/White
 - **Registration marks**
Select **Western**.

- **Scale**
Set to **82%**.
 - **Print Only to File**
Clear this checkbox if it is marked.
6. In the Print Book dialog, click the Print button to begin printing.

Also see [Creating a PostScript File for Commercial Printing](#) on page 171.

Sending the Book to the Printer

This step requires coordination between Technical Publications, Purchasing, and the print vendor. Typically, Wise's Purchasing department handles the ordering of manuals: preparing a request for quote, obtaining an estimate, and sending a purchase order to the printer. We are responsible for getting the files and other deliverables to the printer. The printer representative picks up the files.

1. Notify Purchasing of the date you expect to have the file ready.
2. Verify the print vendor with Purchasing. Currently, we use The Northwest Group, (734) 522-6380. Our representative is Tom Graham. However, Purchasing might want to switch printers or request competitive bids.
3. Give Purchasing a final page count to use on the request for quote and purchase order.
4. WFVI Reference only. Currently, the WfVI Reference manual is perfect bound, which means the cover is all one piece. The cover size depends on the number of pages in the book. Call the printer with the final page count and ask them to estimate the cover size. Give that information to Graphic Design.
5. Gather the PostScript file and other deliverables listed in [Deliverables to the Printer](#) on page 171.
6. Call the printer representative to arrange for pickup of the file.

Also see [Creating a PostScript File for Commercial Printing](#) on page 171.

Reviewing the Printer's Proof

When you first place an order to print a book, you must provide a hard copy to the printer to use as a reference. Be sure to ask the printer to provide a laser or "blueline" proof and return it with the hard copy you printed. When you receive the proof, compare it to your hard copy. If the printer does not return your original hard copy, compare the proof to the final .PDF file of the same book.

At this stage, you are not proofreading the book itself. You are looking for the following:

- ◆ Is the title page correct?
- ◆ Are all pages present and in the correct order?
- ◆ Do the pages break correctly?
- ◆ Is the type style correct?
- ◆ Did all the images print correctly?
- ◆ Are the pages trimmed to the correct size and are they aligned correctly?
- ◆ Are there any extraneous marks on the pages? Typically, small specks are caused by dust on the proof and will not appear in the final printing, but if you're in doubt, point it out.

- ◆ If you happen to find an egregious error in the text, fix it if possible, keeping in mind that the printer will charge extra for changes at this stage.

Mark any problems on the proof and discuss them with the printer.

Also see [Creating a PostScript File for Commercial Printing](#) on page 171.

Creating a .PDF File for Quick Printing

We use quick printers (example: Allegra, Kinko's) for short run printing of user manuals. We use West Wind Litho (Altiris' printing supplier) to print getting started guides.

Quick printers prefer to print from a .PDF file, though they will also accept PostScript. We can't use the online reference .PDF file that ships with the product because it's formatted as 8 1/2 x 11 and our printed manuals are formatted as 5.5 x 8.5 or 7 x 9. To produce a .PDF for quick printing, you first create a PostScript file, because that lets you output in the reduced size. Then you distill the PostScript file to .PDF. Finally, you replace the first placeholder page in the manual with the title page provided by Wise's Graphic Design.

When you order the books, use the [Specifications for Quick Printing](#) on page 178.

Before you begin:

- ◆ Verify that you have a .PDF of the black and white title page from Wise's graphic design department. See [Deliverables from Wise's Graphic Design](#) on page 144.
- ◆ Verify that Generic PostScript Printer appears in the list of printers in Print Setup. If not, see [Installing the Generic PostScript Printer Driver and Setting It Up](#) on page 172.
- ◆ If you are preparing a file for West Wind Litho, verify that you have copied their Distiller job options. See [Setting Up Acrobat Distiller](#) on page 166.

Also see [Finishing Steps Checklist](#) on page 153.

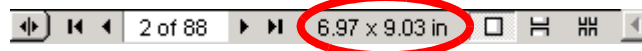
To create a .PDF file for quick printing:

1. From FrameMaker, create a PostScript file (.PS), as described in [Saving the Book as PostScript](#) on page 172. This typically is located in the PDFs_PostScripts/product PDF directory under the book directory. Single-sourced books typically have an additional directory level.
2. Launch Acrobat Distiller.
3. In **Job Options**:
 - For printing at Allegra, Kinko's or other quick printer, select **Press**.
 - For printing at West Wind Litho, select **Win-Acr4-Preps**.
4. Open Windows Explorer and find the .PS file. If there is already a .PDF in this folder, the current distillation will replace it, so rename or move it, or move the .PS file.
5. Double-click the .PS file.

Distiller converts the .PS file to .PDF. This can take several minutes for a large book like Wise for Windows Installer.

If this operation fails and a .PDF is not created, check your PostScript settings. See [Installing the Generic PostScript Printer Driver and Setting It Up](#) on page 172.

6. Open the resulting .PDF in Acrobat and check the following:
 - Make sure the correct fonts are used and are smooth instead of jagged.
 - Spot-check pages throughout the book and compare them to the hard copy (see [Printing a Hard Copy for Proofing](#)). Differences in the pagination could be caused by a font problem.
 - Make sure the page size shown in the lower left of the Acrobat window is correct. (For 7x9 books, this will be a close fraction.)



7. Select Document > Replace Pages.
The Select File With New Pages dialog appears.
8. Select the title page .PDF you obtained from the graphic design department, and then, in the Replace Pages dialog, choose to replace page 1 to 1 of the book .PDF with page 1 of the title page .PDF.

Note

The title page for single-sourced books (example: SoftwareManager) differs depending on whether you are shipping it with Wise Package Studio or WfWI. Make sure you use the correct cover.

9. Save the .PDF file.
10. Move the .PDF file from the PDFs_PostScripts\product PDF directory to the PDFs_PostScripts\quick print PDF directory.

Also see [Delivering Files to West Wind Litho on page 179](#).

Specifications for Quick Printing

Use the following specifications for completing a job ticket for printing manuals at a quick printer.

5.5x8.5 Format Books

For the book pages:

File type provided: PDF
 Color: black & white
 Paper type: 60# white
 Paper size: 8 1/2 x 11
 Sides to print: double-sided
 Collation: collated
 Folding: fold to 5.5 x 8.5
 Binding: saddle stitch

For the book covers:

File type provided: PDF
 Color: full color
 Paper type: 10 pt. coated 1 side
 Paper size: print on 8 1/2 x 11
 Sides to print: single sided
 Folding: fold to 5.5 x 8.5
 Binding: saddle stitch - bind with the pages

7x9 Format Books

For the book pages:

File type provided: PDF
 Color: black & white
 Paper type: 60# white
 Paper size: 8 1/2 x 11
 Sides to print: double-sided
 Collation: collated

Cutting: trim to 7 x 9 (image is centered on page)
Binding: coil binding

For the book covers:

File type provided: PDF
Color: full color
Paper type = 10 pt. coated 1 side
Paper size: print on 8 1/2 x 11
Sides to print: single sided
Cutting: trim to 7 x 9 (image is centered on page)
Binding: coil binding - bind with the pages

Also see [Creating a .PDF File for Quick Printing](#) on page 177.

Delivering Files to West Wind Litho

West Wind Litho is Altiris' printing supplier for product documentation. Currently, we use them to print getting started guides.

Note

This information is sketchy at present because we have only begun to work with West Wind Litho. I will update this as I get more information. JK.

Installing and Setting up an FTP Client

1. Install the FTP client of your choice on your computer. A free FTP client is available from West Wind Litho; you can install it from Technical Publications\Admin\Documentation Software\FTP Commander.
2. Set up an account for West Wind Litho in the DFTP client, using the following information:
 - FTP address: ftp.westwindlitho.com
 - User ID: upload
 - Password: files

Uploading Files

1. (Optional.) For easier uploading, put copies of the files in a single directory.
2. In the FTP client, connect to West Wind Litho.
3. Upload the files to the Wise folder. This procedure will vary depending on the FTP client you use.
4. E-mail West Wind Litho (donna@westwindlitho.com) to notify them of the upload and provide the printing specs.

Also see:

[Creating a .PDF File for Quick Printing](#)
[Specifications for Quick Printing](#)

Chapter 12

Creating Help With WebWorks

This section describes how to prepare FrameMaker files for WebWorks. It includes instructions on using WebWorks Publisher and WebWorks AutoMap. It also describes how we import the WebWorks help files into Visual SourceSafe.

Topics include:

- ◆ [Preparing Frame and .H Files for WebWorks](#)
- ◆ [Using WebWorks Publisher](#)
- ◆ [Installing WebWorks AutoMap](#)
- ◆ [Using WebWorks AutoMap](#)
- ◆ [Importing Help Files into Visual SourceSafe](#)
- ◆ [Adding Bread Crumbs to a Help File](#)

Preparing Frame and .H Files for WebWorks

WebWorks uses our Frame and .H files to generate help files for our different products. The Frame and .H files must be set up correctly to generate the desired help.

This section covers the following:

- ◆ [Setting Up the "Copyright and trademarks" Popup](#)
- ◆ [Setting Up HelpHeader and HelpFooter Markers](#)
- ◆ [Preparing Graphics](#)
- ◆ [Preparing Headings](#)
- ◆ [Connecting Special Links](#)
- ◆ [Mapping Context-Sensitive Help](#)

Setting Up the "Copyright and trademarks" Popup

1. On page 2 of Front Page.fm:
 - Apply the style Z FrontCopyright to the first paragraph under the "Copyright" heading.
 - Apply the style Z2 FrontCopyright to the second paragraph under the "Copyright" heading and to all paragraphs under the "Trademarks" heading. These paragraphs are the text insets, ti_copyright.fm and ti_trademarks.fm, that already have the Z2 FrontCopyright style applied.
 - At the beginning of the first paragraph, add a marker.
Marker Type: Hypertext
Marker Text: newlink copyright
2. On the first page of the Welcome chapter, add the text `Copyright` and `trademarks` to the end of the topic. Apply the style AlsoSee and apply the **help navigation** conditional text tag.
3. At the beginning of the "Copyright and trademarks" text line, add a marker.
Marker Type: Hypertext
Marker Text: gotolink Front Page.fm:copyright
4. If the Frame files to which you add a "copyright and trademarks" popup use a new WebWorks template, you must modify the template settings that determine the background color of the popup. The background color should match the background color of the footer. The background color of the footer is set in document.css of each project template. The background color of the popups in WPS Help is #CCCCCC and #dedfe7 in Wise Help and WPS Web Help. The file in which you set the background color for popups varies depending on whether the template is for HTML help or Web help.

Help	File name	Directory	Value that controls the color
HTML	popup.js	<template>/Support/Scripts	var BackgroundColor
Web	setting.txt	<template>/Support/wwhelp	"background" under the heading <Popups>

The templates are in Technical Publications\Help\WebWorks Templates.

Setting Up HelpHeader and HelpFooter Markers

1. In the Welcome chapter, at the beginning of the first paragraph in the chapter, add a marker to provide the contents of the help header.

Marker Type: HelpHeader

Marker Text: Enter the name of the book. Examples: WiseScript Express or Wise for Windows Installer.

For a list of the book names, see [Inputs Tab](#) on page 205. The names in the Display Name column of the table correspond to the book names.

2. In the Welcome chapter, at the beginning of the second word in the first paragraph, add a marker to provide the contents of the help footer.

Marker Type: HelpFooter

Marker Text: <year> Wise Solutions, Inc. All Rights Reserved.

where <year> is the same year or years as in the FrontPage file of the book.

3. In a chapter that will become stand-alone help, add the HelpHeader and HelpFooter markers as you do in the Welcome chapter. Apply the **standalone help** conditional text tag to the markers only. See [Stand-alone Help from a Single Chapter](#) on page 196.

Preparing Graphics

Always “shrink wrap” anchored frames to their graphics (select the frame and press Esc+m+p). Any margins left in the anchored frame become part of the graphic when it is converted for help.

Where needed, add size markers to graphics. Existing graphics already have size markers, so you only need to add them for new graphics. It is best to do this after you build a preliminary help project for the book, so you can see which graphics need to be resized. For details on adding size markers, see [Sizing Graphics for Help](#) on page 105.

If a graphic should be converted to .JPG format instead of the default .GIF format, add a marker to the graphic frame. See [Specifying the Format for Graphics in Help](#) on page 103.

Preparing Headings

As a general rule, do not skip heading levels. However, if you do skip a heading level, AutoMap automatically collapses the TOC. This is accomplished by adding a property on the Custom Project Properties dialog with a **Name** of TOCCollapse and a **Value** of true; see [Inputs Tab](#) on page 205.

Make sure headings do not contain special characters (examples: ™, ©, or ®). Characters such as hyphens, commas, periods, and apostrophes are OK. In a Web application like Application Gateway, do not use parentheses; see [Troubleshooting AutoMap and Publisher](#) on page 212.

Add a TopicAlias marker to the end of each heading that is mapped to a help ID; see [Mapping Help IDs](#) on page 184.

Connecting Special Links

- ◆ Format URLs as hyperlinks as described in [Creating Links to URLs](#) on page 106.
- ◆ Format SDK links as hyperlinks as described in [Creating Links to External Help Topics](#) on page 107.

- ◆ Use the X-book cross-reference style for cross-book references. Only cross-reference to books that will be installed in the same directory on the destination computer. See [Cross-References to Another Wise Book](#) on page 29.

Mapping Context-Sensitive Help

Wise help is context-sensitive. When the user presses F1 within the product, the appropriate help topic appears. Setting this up involves:

- ◆ Finding the ID for each screen and dialog in the product, see [Finding Map IDs by Changing the Registry](#).
- ◆ Finding the appropriate help topic for each screen and dialog.
- ◆ Adding a line to the help project's Map ID file (.H).
- ◆ Adding a TopicAlias marker to the heading in the Frame file.

Unlike RoboHelp, WebWorks does not require an alias file (.ALI); it generates one internally.

See:

[Where to Store Map ID Files](#)

[Mapping Help IDs](#) on page 184

[Tips for Mapping Specific Kinds of IDs](#) on page 185

Where to Store Map ID Files

To avoid confusion about which Map ID file (.H) to edit for a project, we store a master .H file for each book in a subdirectory named Help ID Master, which is under the book directory. When you need to edit an .H file, edit the one in that directory. We have single-sourced the .H file for WfWI and WIE; it is in the Windows Inst directory. We have also single-sourced the .H file for WiseScript Express, WIS, and WSE; it is in the WiseScript Shared directory.

When you create a help project in WebWorks Publisher or AutoMap, you specify the .H file that's in the Help ID Master directory. Each time you generate the help project, WebWorks uses the master ID file. Stand-alone help files like Debugger.chm for WfWI use the same .H file as the parent product.

Finding Map IDs by Changing the Registry

Development has provided a registry key that, when activated, displays Map IDs. If you need to find just a few IDs, you can do so using RoboHelp; see [Finding Map IDs Using RoboHelp](#) on page 184.

To set the registry key:

1. Open Regedit.
2. Navigate to HKEY_CURRENT_USER\Software\Wise Solutions.
3. Right-click Wise Solutions and select New > DWORD value.
4. Name the value HelpID.
5. In the right pane, double-click HelpID. In **Value data**:
 - Enter 1 to activate this function.
 - Enter 0 to deactivate this function.
6. Click OK and close Regedit.

Shortcuts for Setting the Registry Key

The following .EXEs enable and disable this registry key:

- ◆ TurnOnHelpID.EXE
- ◆ TurnOffHelpID.EXE
- ◆ RemoveHelpIDKey.EXE

These .EXEs are in Technical Publications\Admin\Documentation Software\Turn Help IDs On and Off. Create shortcuts on your desktop if you plan to use them often.

To display Map IDs:

1. Activate the registry key.
2. Start the Wise application.
3. Open a window or dialog and press F1. The Map ID is displayed and is copied to the clipboard, so you can easily paste it into the Map ID file.

Finding Map IDs Using RoboHelp

You can use RoboHelp's HTML BugHunter to find IDs without modifying the registry.

1. Open RoboHelp HTML and create a dummy help project.
2. Select Tools > Enable BugHunter.
3. Click the BugHunter tab in the lower pane.
BugHunter Enabled should appear in this pane.
4. Open the product, find the screen for which you want the ID, and press F1.
If the ID is correctly mapped, the correct help topic appears.
5. Go to the BugHunter pane.
 - If the Map ID is mapped, the "Map Number" line displays the Map ID.
 - If the Map ID is not mapped, the "Results" line displays Map ID in red with a message that it cannot find it.

Mapping Help IDs

Mapping context-sensitive help consists of creating a Map ID text file (.H) and adding TopicAlias markers to the Frame files.

When working on a new release of an existing product, use the Map ID file from the previous release as a starting point. Because large portions of the product usually don't change, a large portion of the Map IDs will be the same. However, be aware that sometimes map IDs change even when the feature itself does not change.

Two or more people can map IDs at a time. Divide the product into sections to minimize duplication of efforts. Make a copy of the Map ID file for each person. When the individuals finish the mapping, merge the separate files into the master Map ID file.

Note

Use comments to group sections in the Map ID text file; precede the comment lines with 2 slashes (//). This helps organize the entries and can help during troubleshooting. It is also helpful when you use the Map ID file for a future release.

An .H file for a single-sourced document can have multiple entries with the same Map ID as long as each is mapped to a unique topic and only one of these topics appears in any

given help project. Because we no longer create WFVS help, this should no longer happen.

To map help IDs:

1. Activate the registry key that enables the display of Map IDs; see [Finding Map IDs by Changing the Registry](#) on page 183.
2. Open the existing Map ID file (.H) or open a new document in Notepad and open the book in FrameMaker.
3. In the product, go to the screen or dialog to link from.
4. In the Frame file, find the topic to link to and copy the topic heading.
5. In the Map ID file, start a new line, type `#define` and a space, and paste the topic name. Replace all spaces and special characters in the topic name with underscores.
6. In the product, press F1 in the screen or dialog you're linking from.
This displays the ID and copies it to the clipboard. If an ID does not appear, you might have to use a different method to find it; see [Tips for Mapping Specific Kinds of IDs](#) on page 185.
7. In the Map ID file, at the end of the line you just created, press Tab and paste the ID.
8. Your new line should look like this:
`#define Working_with_Summary_Items<tab>595`
where `<tab>` is a Tab character.
9. Now copy the topic name, with underscores, from the Map ID file.
10. In the Frame file, at the end of the topic heading line, create a TopicAlias marker. If a marker already exists, make sure it is current.
 - Click to the right of the heading text.
 - Select Special > Marker.
 - In the **Marker Text** field in the Marker dialog, paste the topic title. This must match the topic in the Map ID file exactly.
 - Make sure the **Marker Type** is **TopicAlias**. If it is not, press Shift+Tab to access that field, then type `T` and press Enter.
 - Click the New Marker button.
11. Repeat these steps for every screen and dialog in the product.

Tips for Mapping Specific Kinds of IDs

Mapping F1 Help in the Setup Editor Tables Tab

Following is the expected behavior when a user presses F1 in the Tables tab in Setup Editor:

- ◆ If no table is active, display the top level topic about tables from the Wise help.
- ◆ If a table is active, display the appropriate SDK help topic for that table. If the SDK topic cannot be found, display the top level topic about tables from the Wise help.

Development must program the F1 link to the SDK topic, but this sometimes gets broken between releases. During help development, verify that this is working.

Mapping IDs for WfWI/WIE Inline Help

Wise for Windows Installer and Windows Installer Editor display inline help pages, in HTML format, in Setup Editor. The inline help pages contain links to Wise help topics; these links must be mapped.

If necessary, we can change the IDs in the inline help; just edit the HTML file. Check with Development to make sure the ID you plan to use is not used elsewhere. The document WfWI Inline Help.xls in V:\Technical Publications\Admin\Notes Memos + Misc contains a list of the .HTM files and their IDs. If you change an ID, make sure you change it in this document as well.

For instructions on mapping inline help, see [Linking to Online Help From Inline Help](#) on page 141 and [Linking to SDK Help From WfWI Inline Help](#) on page 142.

Finding IDs for Wise Custom Script Actions

1. Open the Wise Installation System.
2. Select File > Open.
3. Navigate to the Actions directory under the Wise application directory, then select the custom action (.WSE) file to open.
4. Go to Script Editor.
5. From the Event drop-down list, select Exit.
6. Double-click on the Custom Dialog line. The Custom Dialog Editor opens.
7. Click on the Help button in the displayed dialog. Then right-click on the Help button and select Control Properties from the right-click menu.
8. The ID is displayed in the **Display Help Context** field in the Push Button properties dialog.

Application Gateway Map IDs

Application Gateway has a global.asa file with the following section heading, HELP TOPICS FOR WISE DOCUMENTATION TEAM ONLY - DO NOT ALTER!

Caution

Do not make any changes to this file outside of the Wise Documentation Team section. Make changes in this section only if you are sure you know what you are doing.

Under this heading is a list of equations. Only the developer changes the left side of the equation, because changing it requires other development changes. The documentation team changes the right side of the equation when needed. The first 2 equations are for the main help file (an HTM file) and the context sensitive help (the name of the project). The rest of the equations are for the topics that have context-sensitive links. Only topics that are accessed by means of context-sensitive links are included in the global.asa file. Make changes to the right side of these equations as follows:

- ◆ If the name of a topic that has a context-sensitive link changes, update the name of the HTML file in the quotation marks on the right side of the equation.
- ◆ If a new topic, which is accessed by a context-sensitive link, is added, then a new equation must be added to the global.asa file. This must be done in conjunction with the developer, who is responsible for the left-side of the equation.
- ◆ If a topic is no longer accessed via a context-sensitive link, then the equation for that topic can be removed from the global.asa file.

After you update the global.asa file and notify the developer of any changes that they need to make, then check the global.asa file back into VSS.

Note

Management Reports also has a global.asa file, but currently (12/08/03) it does not have a section for the documentation team to update.

Using WebWorks Publisher

We use WebWorks Publisher primarily to create and edit our WebWorks templates. We use WebWorks AutoMap to create, edit, and run jobs that generate help.

A current .EXE file for WebWorks Publisher is in Technical Publications\Help\WebWorks License Keys and Program Files.

See:

- [Setting WebWorks Publisher Preferences](#) on page 188
- [Creating and Editing WebWorks Templates](#) on page 189
- [Adding New Mappings to Project Mappings](#) on page 189
- [Generating Help Using WebWorks Publisher](#) on page 195
- [Specifications for WebWorks Publisher Help Projects](#) on page 196
- [Using WebWorks AutoMap](#) on page 201

Setting WebWorks Publisher Preferences

After you install WebWorks Publisher, set the Preferences.

1. Start WebWorks Publisher.
2. Select File > Preferences.
3. Click the Options tab and set the options as follows:
 - In the **Default Folder for New Projects** field, specify V:\Technical Publications.
 - In the **Default Folder or Share for Templates** field, specify V:\Technical Publications\Help\WebWorks Templates.

Note

This setting gives you access to the different templates that we have created for WebWorks.

- Clear **Automatically display project launcher dialog on startup**.
 - Mark **Remove temporary files when a project is closed or application exits**.
 - Leave **Reset/update FrameMaker version and API client** cleared.
4. Click OK.

WebWorks Templates

We use WebWorks Publisher to create and edit our WebWorks templates. WebWorks AutoMap uses these templates to generate our help.

For HTML help (.CHM files), we purchased a template from WebWorks and edited it to create all our HTML help templates. For our Web help templates (non-compiled help), we edited the WebWorks Help 4.0 template that came with WebWorks Publisher.

For information on the differences between our templates, see:

- [Wise Help \(Template\)](#) on page 191
- [WPS Help \(Template\)](#) on page 191
- [WPS Web Help \(Template\)](#) on page 192
- [WB Inline Help \(Template\)](#) on page 194

Template Locations

- ◆ Active templates are in Technical Publications\Help\WebWorks Templates
- ◆ Archived templates are in Technical Publications\Help\WebWorks Templates Archive
- ◆ WebWorks also provides a set of templates in a Template directory of both AutoMap and Publisher.

All WebWorks templates have a .WFP file and a Support directory.

Creating and Editing WebWorks Templates

To create a new template, copy and edit an existing template. To edit a template, open its .WFP file in WebWorks Publisher. Use the WebWorks Publisher dialogs below to edit templates. For details on using these dialogs, see the WebWorks Publisher Help. To add new mappings to the project mappings, see [Adding New Mappings to Project Mappings](#).

Dialog Name	Use to:
Style Designer	create or edit WebWorks styles
Project Mappings	map the Frame styles to the WebWorks styles
Project Options	set the options for the project

Note

Before you can map a Frame style to a WebWorks style in the Project Mappings dialog, you must first create the WebWorks style using the Style Designer.

After each release, we archive the WebWorks templates to the WebWorks Templates Archive directory; see [Locating WebWorks AutoMap Files](#) on page 201.

Adding New Mappings to Project Mappings

When you add new tags (paragraph, character, or table) or markers to a FrameMaker template, these tags and markers must be added to the WebWorks template and mapped to the appropriate WebWorks styles. To do this, you must work through a procedure similar to [Generating Help Using WebWorks Publisher](#) on page 195.

Note

Before you perform this procedure, archive the WebWorks template.

1. Open WebWorks, select File menu > New, and select the template.
If the Wise templates do not appear, change the default folder for the templates; see [Setting WebWorks Publisher Preferences](#) on page 188.
The New Project Wizard - Step 1 dialog appears.
2. Complete the Step 1 dialog:
 - **Please chose a name for this project**
Because this is a temporary project, any name will do.
 - **Please choose a folder location**
Specify a temporary location for creating this project.
3. Click Next.

- The New Project Wizard - Step 2 dialog appears.
4. Specify a file or files to be converted that contain the new tags or markers.
If a tag or marker is not in the files, then it will not appear in the WebWorks Publisher Project Mappings dialog, and you will not be able to map it to a WebWorks style. Because you will need to compile this project to test your changes, select as few files as possible to speed up the process.
 5. Click Next.
The New Project Wizard - Step 3 dialog appears.
 6. If you are mapping the new style to an existing WebWorks style, locate the new style in the Source Style column and make the appropriate selections in the "WebWorks Style," "Split," and "In TOC?" columns.
If there is a WebWorks style with the same name as the source style, it is mapped automatically. If the WebWorks style does not exist, you will have to create it first before you can do the mapping. See below.
 7. Click Next on the Step 3 dialog.
The New Project Wizard - Step 4 dialog appears.
 8. Because each AutoMap job has its own conditions, click Next without making any changes.
The New Project Wizard - Step 5 dialog appears.
 9. Click Next on the Step 5 dialog.
The New Project Wizard - Step 6 dialog appears.
 10. Mark **No, I'll convert the content later** and click Finish.
A new project opens in WebWorks.
 11. To create a new WebWorks style, select Style menu > Designer.
The Style Designer dialog appears. Use it to create new styles and click OK. See WebWorks help for details.
 12. To map a FrameMaker tag or marker to a new WebWorks style, select Project menu > Mappings.
The Project Mappings dialog appears. Use this dialog to map the style and click OK.
 13. Generate the project by selecting Project > Generate All.
 14. If you see a message that errors were found, click OK.
Ignore all broken cross-reference errors. However, red error messages must be resolved before you can use this template to update the original template. See [Troubleshooting AutoMap and Publisher](#) on page 212.
 15. Click the Files tab and do one of the following:
 - If you created a .CHM, double-click it in the HTML Help Files folder.
 - If you created Web Help system, double-click the first HTM file in the WebWorks Help Files folder.
 16. Verify that the changes you made to the template appear in the help.
If the new mappings do not work correctly, continue to edit them until they do. When you are done editing, you have a new template that continues your new mappings.
 17. To replace the template you selected at the beginning of this project with this new template, select File menu > Save as Template.
The Save As Template Wizard Step 1 dialog appears.

18. Click Next.
The Save As Template Wizard Step 2 dialog appears.
19. Complete the dialog:
In the first field, enter the name of the template you are modifying. (Example: Wise Help.) In the second field, specify the V:\Technical Publications\Help\WebWorks Templates directory.
20. Click Finish.
A warning appears that this file already exists.
21. Click Yes to overwrite the old template in the WebWorks Templates directory.

Wise Help (Template)

This template is used for Wise for Windows Installer, Wise Installation System, and Wise products other than Wise Package Studio.

- ◆ All heading text is black, which is the default.
- ◆ In document.css, the following colors are defined. This file is located in the template's Support directory. Edit it with Notepad.

```
a:active { color: #FF0000}
a:hover  { color: #FF0000}
a:link   { color: #0000FF}
a:visited { color: #800080}
background-color: #FFFFFF
H1 Background-color: #98CBFF
Header background-color: #98CBFF
Footer background-color: #CCCCCC
```

Sample Script Help

When we use the Wise Help template to create the Sample Script help for WiseScripts, we make the following changes to the template:

- ◆ Remove all splits from the Project Mappings to create a single HTML file that includes all the topics.
- ◆ In the Normal.asp file, remove the entire <tr> tag that contains \$BP80PageASP_Navigating. This removes the navigation buttons at the bottom of the topic.
- ◆ Changed the default Extension to HTM. This is done in the Style Designer on the Options tab of the Pages tab.

Wise SDK (Template)

This template is used for the Wise SDK.

- ◆ It is based on the Wise Help template.
- ◆ The Copyright marker in the Style Designer has the following action function: @SET_GLOBAL(copyright, \$DATA; Created \$DATE(B);\, \$DATE(Y);.);. This function adds the month and year when the CHM file is created.

WPS Help (Template)

This template is used for Wise Package Studio and its components.

- ◆ The value for the color of the ChapterTitle, Heading1, Heading2, and Heading3 text is #185173. In the Style Designer dialog, click the Paragraphs tab at the top, and then

click the Font tab at the bottom. Select **ChapterTitle** from the **Name** drop-down list. Enter #185173 in the **Color** field. Repeat for Heading1, Heading2, and Heading3.

- ◆ In document.css, the following colors are defined. This file is located in the template's Support directory. Edit it with Notepad.

```
a:active { color: #0000CC}
a:hover  { color: #CC0033}
a:link   { color: #3366CC}
a:visited { color: #800080}
background-color: #FFFFFF
H1 Background-color: #c0cbd1
Header background-color: #c0cbd1
Header color: #185173
Footer background-color: #dedfe7
```

WPS Web Help (Template)

WPS Web Help is a custom template created by modifying the WebWorks Help 4.0 template that comes with WebWorks Publisher. It generates non-compiled help files that can be used with Web applications such as Application Gateway and Management Reports.

Note

If you reference the same graphic more than once in the Frame files, use a .JPG instead of a .TIF. Each time you reference a .TIF, a separate .GIF is created in the help files.

It generates one .HTM file and one .HTML file for each topic. The .HTM file is the entry point to the HTML files. To change the name of the .HTM file:

- ◆ In WebWorks AutoMap, specify the name of the .HTM file on the Custom Project Properties dialog; see [Inputs Tab](#) on page 205.
- ◆ In WebWorks Publisher, open the Project Options dialog and select the WebWorks Help tab (appears only when you are working with a Web Application template), and specify the name of the file in the field at the bottom of the tab. If you are creating the help with WebWorks AutoMap, you don't need to modify this field.

We designed the WPS Web Help template to create, as much as possible, help that has the same look as our Wise Package Studio help. Example: The color settings in document.css are the same for both templates. However, when we created this template, we had to make the following template changes:

Style Designer Settings

Because Web help uses different technology than HTML help, the code for its macros is slightly different. It is important to be aware of these differences when creating or editing a Web help template, especially when copying code from an HTML template to a Web help template. Access the macro code in the Style Designer dialog by clicking the Paragraph tab and then clicking the Macro tab.

The macro code in Web help differs from that in the HTML help as follows:

- ◆ The following 4 lines of code are inserted. The first line appears at or near the beginning of every macro and the second through fourth lines appear at the end of the macro:

```
@SET_GLOBAL(var_PopupText,
); \
\
$BP80Popups_ParaMacro;\
```


Caution

When you create new styles, this code should appear automatically. However, you need to be careful not to delete these lines when copying and pasting a macro from another template. Example: If you copy a macro from an HTML template, it does not have these lines of code. When you paste the macro, don't paste over these lines of code.

- ◆ In the Edition2, Notebody, Notebody2, Notebody3, Notehead, and Notehead2 styles, `<hr width="100%" size="1" noshade="noshade" />` replaces `<hr width="100%" size="1" noshade>`. Be sure to include a space after "noshade." XML requires these changes.
- ◆ In the ChapterTitle, Header1, Header2, and Header3 styles, `<p class="Header">` replaces `<p class=Header>`. XML requires the quotation marks.

Document.css

We created the WPS Web Help document.css file (located in the Support directory) by modifying the document.css file that came with the WebWorks Help 4.0 template. The following changes were made to this file:

- ◆ We added the following block of code after the background-color block to adjust the margins:

```
blockquote {
margin-left: 8px;
margin-right: 12px;
}
```

It would be better if both margins were the same size, and according to their Help, they should both be 10px. However, these are the settings WebWorks uses in its own Web help to keep the right edge of the text within the window when a topic has a vertical scroll bar.

- ◆ We inserted the following code at the end of the WPS Web Help document.css file: All of the code in the WPS Help template document.css file from H1 to the end of the file.
- ◆ We changed the Footer margins as follows:

```
margin-left: 6.00 pt;
margin-right: 9.00 pt;
```

This aligns the margins of the footer with those of the header.



Normal.asp

We modified the WebWorks Help 4.0 Normal.asp (located in the Support folder) as follows:

- ◆ Removed the two blocks of code that deal with the logo and the company information along with `<hr align="left"/>` and `<hr/>`.
- ◆ Copied a block of code for the footer (which begins with `<div` and ends with `</div>`) from the normal.asp file of the WPS Help template and inserted it after `</blockquote>`.
- ◆ Made the following changes to this block of code:
 - The word `placeholder` was deleted.
 - Quotation marks were put around the two occurrences of the word `footer` to make this code XML compliant.

Additional Changes

You can make the following changes to the look of the Web help:

- ◆ You can customize the look of the main window by modifying settings in the settings.txt file. This file is in the WPS Web Help\Support\wwhelp directory. (Example: You can change the color of the tabs or the text that appears on the tabs.) For details, see *Editing the Settings.txt File* in the WebWorks help.
- ◆ You can replace the buttons, such as , that appear in the toolbar frame above the topic with customized buttons. The files for these buttons are in the support\wwhelp\wwhimpl\common\images directory. Each button has two graphics. The graphics must be in a GIF format and be 23 X 21 pixels. To turn off one of these buttons or add an e-mail button, select Project menu > Options > WebWorks Help tab. Checkboxes corresponding to each toolbar are on the right.
- ◆ You can replace the splash screen that appears when the help is first opened. Its file is in the support\wwhelp\images directory. It must be a JPG file and be named splash.jpg. There are no restrictions on its size. To include text on the splash screen, edit default.htm, which is in the support\wwhelp\wwhimpl\common\html directory. Currently, our Web help opens to a specific topic and the end user never sees the splash screen unless they directly open the HTM file. The current splash screen for the WPS Web Help template is a picture of the Package Studio icon .
- ◆ There are two files that control the initial width of the navigation frame. They are support\wwhelp\wwhimpl\java\html\wwhelp.htm and support\wwhelp\wwhimpl\js\html\wwhelp.htm. To change the initial width of the navigation frame, locate `<frameset cols = 275, *"` in both of the files and replace 275 with a larger or smaller number. The WebWorks default setting was 300.

WB Inline Help (Template)

- ◆ To create the WB Inline Help's document.css file, we modified the WPS Help document.css file as follows:
 - The background color under Default style was changed to:


```
background-color: #e5eaed
```
 - The following lines were deleted:


```
H1
{
Background-color: #c0cbd1;
}
```
- ◆ We also made the following template style changes in the Style Designer dialog:

Style	Setting Changes
Heading 2	Basic subtab: <ul style="list-style-type: none"> ◆ Vertical Spacing Above: 9.0pt ◆ Vertical Spacing Below: deleted current value and left blank ◆ Vertical Spacing Line: deleted current value and left blank ◆ Markup Tag: h2
Heading 3	Basic subtab: <ul style="list-style-type: none"> ◆ Vertical Spacing Above: 9.0 pt ◆ Vertical Spacing Below: deleted current value and left blank ◆ Vertical Spacing Line: deleted current value and left blank ◆ Markup Tag: h3

- ◆ The following WPS Help template mappings were changed on the Mappings tab of the Project Mappings dialog.

Source Style	WebWorks Style	Split	In TOC?
ChapterTitle	NoOutput	clear	Unmap
H1 Heading1	NoOutput	X	Unmap
H2 Heading2	Heading2	clear	Unmap
H3 Heading3	Heading3	clear	Unmap
XH1 Heading1	Heading2	clear	Unmap
XH2 Heading2	Heading3	clear	Unmap

Generating Help Using WebWorks Publisher

Normally, we use WebWorks AutoMap to generate our help; see [Using WebWorks AutoMap](#) on page 201. If you are working on a small project that doesn't justify creating an AutoMap job or if AutoMap is not working, then use this procedure.

1. If you are doing a final help build, complete at least the help-related finishing steps described in [Finishing Steps Checklist](#) on page 153. If you are making a "draft" help build, you can skip this step.

If this help project contains cross-book references and the referenced help project is not open, you might see some errors relating to those references. Those errors will be resolved when you build the referenced help project.

If you are using the WB Inline template, the following message will appear in the log in blue: "Please map TOC styles!" Ignore this message because we don't want to map TOC styles.

At this point, you can save and close the project. If this book has cross-references to another book, leave this project open while you generate the second project. For a list of which files to leave open, see [Specifications for WebWorks Publisher Help Projects](#) on page 196.

Stand-alone Help from a Single Chapter

To create stand-alone help for a single chapter (example: Debugger), import only that chapter to a new help project. The differences between stand-alone help and a full help project are:

- ◆ The stand-alone help does not contain the copyright popup, because we do not import the Front Page file.
- ◆ Normally, text for the help header and footer is obtained from special markers in the Welcome chapter. Because we do not include the Welcome chapter in standalone help, those markers are in the stand-alone chapter and have the condition "standalone help." Hide the standalone help tag when importing the entire book for the main help, and show it when importing the single chapter for stand-alone help.
- ◆ When you create the stand-alone help project, you specify the single chapter instead of the entire book.

Specifications for WebWorks Publisher Help Projects

Each Wise product has different help requirements (examples: the template used to map styles, conditional text, and the order in which you build the help projects). Use the following sections as guides for creating help for each Wise product.

- ◆ [Creating Wise for Windows Installer Help](#)
- ◆ [Creating Wise Package Studio Help](#) on page 197
- ◆ [Creating WPS Web Help](#) on page 197
- ◆ [Creating Workbench Inline Help](#) on page 198
- ◆ [Creating Wise InstallTailor Help](#) on page 198

Creating Wise for Windows Installer Help

Note

Don't use these specifications when generating help with WebWorks AutoMap.

Template: Wise Help

Build order	Project Name	Show Conditions	Generate and
1. WiseScript Express	Express	help navigation	save and close
2. Stand-alone Debugger	Debugger	help navigation standalone help WfWI only	save and close
3. Software Manager	SoftwareManager	help navigation WfWI only	leave open
4. Wise for Windows Installer	WfWI	help navigation WfWI only	leave open
5. Regenerate Software Manager			

When you finish, save and close all help projects.

Creating Wise Package Studio Help

Note

Don't use these specifications when generating help with WebWorks AutoMap.

Template: WPS Help

Build order	Project Name	Show Conditions	Generate and
1. WiseScript Express	WiseScriptExpress	help navigation WPS	save and close
2. Stand-alone Debugger	Debugger	help navigation standalone help WIE only	save and close
3. Workbench	Workbench	help navigation	leave open
4. ConflictManager	ConflictManager	help navigation	leave open
5. Software Manager	SoftwareManager	help navigation WPS	leave open
6. WiseScript Editor	WiseScriptEditor	help navigation	leave open
7. Windows Installer Editor	WindowsInstallerEditor	help navigation WIE only	leave open
8. Regenerate in this order:			
◆ Workbench			
◆ ConflictManager			
◆ Software Manager			
◆ WiseScript Editor			
◆ Windows Installer Editor			

Note

When you regenerate each project, you should not see any cross-reference errors on the Log tab.

When you finish, save and close all help projects.

Creating WPS Web Help

WPS Web Help creates HTML files that are used by Web applications such as Application Gateway and Management Reports. Instead of a compiled .CHM file, it creates a help system made up of many files.

Note

If you reference the same graphic more than once in the Frame files, use a .JPG instead of a .TIF. Each time you reference a .TIF, a separate .GIF is created in the help files.

Deliverables:

The entire contents of the Output directory.

Normally, you don't have to deploy any files. When the help is built, any files that have changed are imported into Visual Source Safe by our automated import process. If you need to deploy the deliverables, use the WebWorks Deployment Wizard.

1. Select File menu > Deploy/Merge.
The Deployment Wizard dialog appears.
2. Select a target destination from the list. If the desired target destination is not listed, click Add to define it.
This is the location where you want to deploy the Web help you have created.
3. Click Next.
The Choose Project(s) dialog appears.
4. Select the WPS Web Help project and click Next.
The Merge Settings dialog appears.

Note

You cannot deploy .CHM help files. When you try to deploy them, you will get error messages.

5. Click Next.
6. Click Finish on the final dialog to publish the deliverables to the desired destination.

Template:

WPS Web Help

Build Order:

No build order is required because there is only one help project with no cross-book references.

Also see [WPS Web Help \(Template\)](#) on page 192.

Creating Workbench Inline Help

Note

Don't use these specifications when generating help with WebWorks AutoMap.

Deliverables:

The following files from the Output directory: all the HTML files, document.css and catalog.css, and the data and scripts folders.

Template:

WB Inline Help (This template was created from the WPS Help template.)

Build Order:

No special build order required.

Creating Wise InstallTailor Help

This is a very small help file, therefore, we do not use AutoMap to generate it.

Deliverable: InstallTailor.chm, from the Output directory.

Template: Wise Help

Build instructions: No special build instructions. See [Generating Help Using WebWorks Publisher](#) on page 195.

Generating an Alias File

This step has become obsolete because with WebWorks an alias file is automatically created. However, we should retain this procedure in case we change our procedure for creating help and again need to create this file.

Our Development department has created a utility that generates an alias file by scanning the project directory and its subdirectories for all .HTM files and writing the file names to a text file in the proper syntax.

Normally, the alias file contains one line for every line in the Map ID file. Our utility, however, creates a line for EACH file in the help project. The advantage of building the alias file this way is that you won't have to update the alias file every time you add a new ID and topic to the Map ID file—the alias file already contains aliases for ALL topics. It is not a problem to have extraneous entries in the alias file.

1. Open Technical Publications\Admin\OLD Help Templates\OLD Robo HTML templates + files\HtmlHelpUtility.exe.
2. In **Scan directory**, specify the root help project directory.
3. In **Text file path**, specify the name and location of the alias file that will be created. Save it in the root help project directory, and use the same name as the help project file, except with the extension .ALI.
4. Click OK.

The utility program scans the selected directory and creates the alias file.

Installing WebWorks AutoMap

Installing WebWorks AutoMap requires the installation of 3 WebWorks programs. An .EXE file for each of these programs is in Technical Publications\Help\WebWorks License Keys and Program Files. They must be installed in the order specified below. WebWorks frequently updates these .EXE files. To check for updates, go to <http://www.webworks.com/support/downloads.aspx>.

WebWorks Publisher requires the installation of FrameMaker 5.5 or later.

1. Install .NET FrameWork 1.1 or later.
Download it from the Microsoft Download Center.
2. Install WebWorks Publisher by running wpubfm.exe.
3. Install WebWorks License Manager 3.0 by running wlicsrv.exe.
4. Install a license key in the License Manager:
 - Start WebWorks License Manager by selecting Start > Programs > Quadralay > WebWorks License Manager 3.0 > WebWorks License Manager.
 - Go to Technical Publications\Help\WebWorks License Keys and Program Files\AutoMap_license_keys.txt and copy the first license key.
This file contains each key with a brief description.
 - On the WebWorks License Manager dialog, paste the key into **Key Information** and click Add.
 - Go back to AutoMap_license_keys.txt and copy the second key, paste it into **Key Information**, and click Add.
We should only need to use 1 key, but WebWorks gave us an additional key to deal with problems we were having. With the updating versions of AutoMap, this additional key may not be needed.
 - If you have problems with the License Manager, see [Troubleshooting AutoMap and Publisher](#) on page 212.
5. Install AutoMap by running wauto.exe.

To run AutoMap and set its preferences, see [Setting WebWorks AutoMap Preferences](#) on page 202.

Using WebWorks AutoMap

We use AutoMap to create, edit, and run jobs that automatically generate our help files. We use WebWorks Publisher primarily to create and edit our WebWorks templates; see [Using WebWorks Publisher](#) on page 188. AutoMap automates the process of generating help files by running, at a scheduled time, an AutoMap job designed specifically for each of our products. Each job accesses the appropriate Frame and .H files and applies the appropriate Custom Properties, Conditions, and Variables. After AutoMap generates the help files, it outputs them to a specified directory from which they are automatically imported into VSS.

See:

[Locating WebWorks AutoMap Files](#)
[The WebWorks AutoMap Window](#) on page 202
[Setting WebWorks AutoMap Preferences](#) on page 202
[Creating or Editing AutoMap Jobs](#) on page 203
[Updating AutoMap Jobs](#) on page 209
[Scheduling AutoMap Jobs](#) on page 211
[Using AutoMap to Create PDFs](#) on page 211
[Troubleshooting AutoMap and Publisher](#) on page 212
[Importing Help Files into Visual SourceSafe](#) on page 215

Locating WebWorks AutoMap Files

Most AutoMap files are located in \Technical Publications\Help, in the following subdirectories:

WebWorks Jobs

This directory contains all the AutoMap job files in .XML format. It also contains a logs directory for each job. Each time a job runs, another log is added to this directory in .TXT format. Use these logs to troubleshoot AutoMap; see [Troubleshooting AutoMap and Publisher](#) on page 212.

WebWorks License Keys and Program Files

This directory contains an Automation subdirectory with the following files:

HelpBuild.ini
 VSS_Help_Automation.EXE
 VSS_Help_Automation.wse

These files are used to automate the process of checking our help files into VSS; see [Importing Help Files into Visual SourceSafe](#) on page 215.

This directory contains a .TXT file that lists our AutoMap license keys; for information on using these licenses, see [Installing WebWorks AutoMap](#) on page 200.

This directory also contains the latest version of the following WebWorks programs:

- ◆ WebWorks Publisher for FrameMaker: wpubfm.exe.
- ◆ WebWorks License Manager: wlicsrv.exe.
- ◆ WebWorks AutoMap: wauto.exe.

WebWorks Staging Area

This directory contains all the files that AutoMap generates when it runs a job. It puts the files for each job in an Output or Projects directory. After WebWorks generates these

files, it copies some of them to another output directory specified in the AutoMap job; see [Output Tab](#) on page 209.


WebWorks Templates

This directory contains all of our active templates with their support files; see [WebWorks Templates](#) on page 188. To access these templates in AutoMap, specify this directory in AutoMap preferences. We store 2 Altiris templates here, which are copied from \\Docdev\Docs\Doc_Standards\FrameMaker\WebWorks_Templates.

WebWorks Templates Archives

This directory contains all of our archived templates with their support files. After each release archive the templates used for that release to this directory.

Setting WebWorks AutoMap Preferences

1. Start AutoMap by selecting Start > Programs > Quadralay > WebWorks AutoMap 2003 > WebWorks AutoMap 2003 Administrator.
2. Do one of the following:
 - If this is the first time you run AutoMap, a dialog appears asking you to set the template directory. Click OK.
 - Access the Preferences dialog by clicking .


The Preferences dialog appears.
3. Set the preferences as follows unless the location of these folders has changed.
 - In the Template Folder, specify:
V:\Technical Publications\Help\WebWorks Templates
 - In the Staging Folder, specify:
V:\Technical Publications\Help\WebWorks Staging Area
 - In the Job Folder, specify:
V:\Technical Publications\Help\WebWorks Jobs
 - Clear **Keep temporary files from each job**.



You can start creating AutoMap jobs; see [Creating or Editing AutoMap Jobs](#) on page 203.

The WebWorks AutoMap Window

To run AutoMap and set its preferences, see [Setting WebWorks AutoMap Preferences](#) on page 202.

Until you create AutoMap jobs, the AutoMap window contains an empty table. When you create a job, it is listed in the table and the following columns display information concerning that job:


Column	Description
Name	The name you give the job on the Basic tab; see Basic Tab on page 204. This name is merely a means of identifying this job. It does not appear in the help that is generated.
Suspend	This value is always false unless you suspend the job. To suspend a job, select the job and click  .

Column	Description
State	This column is always empty unless the job is running and then it has the value of running. To run a job, by click  or schedule the job to run at a specified time. To cancel a job that has a status of running, click  .
Format	The name of the template you assign to the job on the Basic tab; see Basic Tab on page 204.
Input Source(s)	Lists all the help projects that make up this job. You add these on the Inputs tab; see Inputs Tab on page 205.
Output Destination	The name you assign to the output directory for this job; see Output Tab on page 209.
Conditions	Lists all the conditions with their value for this job; see Conditions Tab on page 207.
Next Run	Specifies when this job is scheduled to run next.
Last Run	Specifies when this job was last run.
Last Run Status	Specifies the status of the job when it last ran. Possible statuses are: <ul style="list-style-type: none"> ◆ OK = Job completed with no errors. ◆ WARNINGS = Job completed with warnings. ◆ ERROR = Job completed with errors. ◆ DEPLOYMENT ERROR = The job process failed. ◆ ABORTED = The job was aborted by the user. ◆ LICENSE ERROR = AutoMap aborted because no licenses were available when the job was run.

Creating or Editing AutoMap Jobs

To start AutoMap and set its preferences, see [Setting WebWorks AutoMap Preferences](#) on page 202. For details on the AutoMap Window, see [The WebWorks AutoMap Window](#) on page 202.

To create a new job:

1. In AutoMap, click .

The first step of the new job wizard appears.
2. Complete each step in the new job wizard.


The steps in the new job wizard correspond to the tabs in Job Properties dialog. Because the Job Properties dialog is used more frequently than the new job wizard, the instructions for creating a job are organized around the tabs of the Job Properties dialog.
3. For details on each step in the new job wizard, see the instructions for the corresponding tab on the Job Properties Dialog:

Step in New Job Wizard	Tab on Job Properties Dialog
Name and Format - Step 1 of 6	Basic Tab on page 204
Input Sources - Step 2 of 6	Inputs Tab on page 205

Step in New Job Wizard	Tab on Job Properties Dialog
Merge Settings - Step 2(*) of 6	Merge Tab on page 206
Condition/Media Type Settings- Step 3 of 6	Conditions Tab on page 207
Global variable settings - Step 4 of 6	Variables Tab on page 208
Output Destination - Step 5 of 6	Output Tab on page 209
Scheduling - Step 6 of 6	Click Schedule on the Basic Tab

- If the Enter Account Information dialog appears when you complete the wizard, enter you network password and click OK.

To create a job by duplicating and editing an existing job:

- Select the job to duplicate and click . The Enter Account Information dialog appears.
- Enter your network password and click OK. The job is duplicated and given the name Job<n>, where n is a number that corresponds to the number of jobs that have Job<n> for a name.
- Double-click Job<n> that you just created. The Job Properties dialog opens.
- Edit the information on each of the tabs of this dialog as needed. See the table above to locate instructions for each tab.

Basic Tab

This tab is on the Job Properties dialog in AutoMap. To open this dialog, double-click the job to edit.

In the following fields, enter general information about the job:

◆ Job Name

Enter a name that clearly identifies this job. This name only appears in the list of jobs in AutoMap.

◆ Template

Select the correct template from the following chart:

Jobs	Template
Application Gateway	WPS Web Help
Management Reports	
QA	WPS Help
WB Inline	WB Inline Help
WfWI	Wise Help
WfWI Debugger	
WiseScript Express	WPS Help
WIE Debugger	
WPS	

For information on the differences between the templates, see [WebWorks Templates](#) on page 188.

- ◆ **Post Deployment Script**
Leave blank. If we create a script to insert the cover art into PDFs, that script will probably go here.
- ◆ **Source Type**
Select FrameMaker.
- ◆ Click the Schedule button to set up the schedule for running this job; see [Scheduling AutoMap Jobs](#) on page 211.

Inputs Tab

This tab is on the Job Properties dialog in AutoMap. To open this dialog, double-click the job to edit.

On this tab, you specify the projects that make up the job. For each project, you specify its Frame files and Custom Properties.

To add or edit a job's projects:

1. On the Inputs tab, click Add or select a project (input source) and click Edit.
The Input Source Editor dialog appears.
2. Enter the **Project Name** and **Display Name** for the project based on the table below.

The **Project Name** is used by default as the name for the .CHM file and the name for the folder in the WebWorks Staging where AutoMap puts generated files. According to WebWorks Help, the **Display Name** appears on the title bar of the help window. Actually, the title bar displays the value in the HelpHeader; see [Setting Up HelpHeader and HelpFooter Markers](#) on page 182.

Job	Project Name	Display Name
Application Gateway	ApplicationGateway	Application Gateway
Management Reports	ManagementReports	Management Reports
QA	QualityAssurance	Wise Package Studio - Quality Assurance
WB Inline	InlineHelp	Inline Help
WfWI	Express	WiseScript Express
	SoftwareManager	Software Manager
	WfWI	Wise for Windows Installer
WfWI Debugger	Debugger	Debugger for Windows Installer
WIE Debugger	Debugger	Debugger for Windows Installer
WiseScript Express	Express	WiseScript Express
WPS	ConflictManager	ConflictManager
	SoftwareManager	Software Manager
	WindowsInstallerEditor	Windows Installer Editor
	WiseScriptEditor	WiseScript Editor
	Workbench	Wise Package Studio

3. Click Custom Properties.
The Custom Project Properties dialog appears.
4. Enter the **Name** and **Value** for each of the Custom Project Properties in your project based on the table below. These properties vary depending on the type of help you are generating. Currently, only Application Gateway and Management Reports use Web help.

Type of Help	Name	Value
HTML	UseCustomMapFile	true
	UseFilenameAsTopicAlias	true
	MapFile	Copy and paste the path and name of the MapID file for this project. Single chapter Jobs (ex. Debugger) use the same MapID file as is used by the larger project to which they belong.
	TOCCollapse	true
Web	EntryPoint	ApplicationGateway.htm ManagementReports.htm
	UseFilenameAsTopicAlias	true
	TOCCollapse	true

5. Click OK to save the projects Custom Properties.
6. When creating a new job, mark **Use the following file or files** and specify the Frame files that make up this project.

Note

Don't specify a .book file, because you need a listing of all the files so you can move Front Page.fm to the end.

7. To delete a file from an existing job, select it and click Delete. To add files to an existing job, specify the files.
If the name of a file changes in a new release, delete it and specify the new file.
8. Use Move Up and Move Down to arrange the files for the project.
Be sure the Front Page.fm file is last. It creates an empty topic that is probably never seen if it is the last topic in the help.
Use the Frame book as a guide to put the chapters in the correct order.
9. Click OK in the Input Source Editor dialog.
The Inputs tab of the Job Properties dialog reappears.
10. If you need to add or edit additional projects repeat the preceding steps.
11. When you finish, click OK in the Job Properties dialog.

Merge Tab

This tab is on the Job Properties dialog in AutoMap. To open this dialog, double-click the job to edit.

This tab is used to merge one or more help files into a master help file. Currently, we don't use this feature, but we can't turn it off. In Master Title, enter Master for the name of the merged help file. The file that is generated with this name is not used.

Conditions Tab

This tab is on the Job Properties dialog in AutoMap. To open this dialog, double-click the job to edit.

On this tab, you add or edit the job's conditions and specify whether they are visible or hidden. When you give conditional text the value of visible, that conditional text appears in your help.

Note

You set conditions for an entire job. You cannot set conditions for an individual project within a job. Example: If you assign the condition WPS the Value of visible, it will be visible in all projects.

To add or edit a job's conditions:

1. On the Conditions tab, click Add or select a condition and click Edit.
The Condition Editor dialog appears.
2. Enter the **Name** and **Value** for each condition in your job as described in the two tables below and click OK.

Condition Name	Application Gateway	Management Reports	QA	WB Inline
comment	hidden	hidden	hidden	hidden
help navigation	visible	visible	visible	visible
MgtReports		visible	hidden	
not for help		hidden	hidden	hidden
QA		hidden	visible	
WPS		hidden	hidden	

Condition Name	WfWI	WfWI Debugger	WPS	WIE Debugger	WiseScript Express
comment	hidden	hidden	hidden	hidden	hidden
Express			hidden		visible
help navigation	visible	visible	visible	visible	visible
MgtReports			hidden		
not for help	hidden	hidden	hidden	hidden	hidden
QA			hidden		
standalone help		visible		visible	
WfWI only	visible	visible	hidden	hidden	

Condition Name	WfWI	WfWI Debugger	WPS	WIE Debugger	WiseScript Express
WIE only	hidden	hidden	visible	visible	
WIS			hidden		hidden
WIS+WSE			visible		hidden
WSE			visible		hidden
WPS	hidden		visible		

Variables Tab

This tab is on the Job Properties dialog in AutoMap. To open this dialog, double-click the job to edit.

On this tab, you add or edit the job's variables. When you assign a value to a variable, that value replaces the Frame variable in your help.

Note

You set variables for an entire job. You cannot set variables for an individual project within a job. Example: If you assign the variable Product_SM the value of Wise for Windows Installer, it will have this value in all projects.

To add or edit a job's variables:

1. On the Variables tab, click Add or select a variable and click Edit.
The Enter name and value dialog appears.
2. Enter the **Name** and **Value** for the variables for your job according to the following table:

Job	Variable Name	Variable Value
MgtReports	Product_WB	Wise Package Studio
QA	Product_WB	Quality Assurance
WfWI	Product_WinInst	Wise for Windows Installer
	Product_SM	Wise for Windows Installer
WfWI Debugger	Product_WinInst	Debugger for Windows Installer
WPS	ProductName	Wise Package Studio
	Product_WinInst	Windows Installer Editor
	Product_SM	Wise Package Studio
	Product_WB	Wise Package Studio
	Product_WiseScr	WiseScript Editor
	ScrEditorUI	Setup Editor
WIE Debugger	Product_WinInst	Debugger for Windows Installer
WiseScript Express	Product_WiseScr	WiseScript Express

Output Tab

This tab is on the Job Properties dialog. To open this dialog double-click, the job to edit. On this tab, you select the destination for the help files that AutoMap generates. You can also create new destinations or edit existing destinations. After AutoMap generates the help files, it copies the files that are checked into VSS from the WebWorks Staging Area to the directory specified on the Output tab.

The WB Inline job is an exception to this rule. It generates a .CHM file that is copied to the directory specified on the Output tab. However, we use the HTML files that remain in the WebWorks Staging Area; see [Locating WebWorks AutoMap Files](#) on page 201.

When the VSS Help Automation program runs, it copies the files from directory specified on the Output tab (with the exception of the WB Inline help files) to another staging area, and then into VSS; see [Importing Help Files into Visual SourceSafe](#) on page 215.

To select a destination for your output:

1. If the desired destination is listed, mark its checkbox.
2. If it is not listed, click Add to create a new destination or select an existing destination and click Edit.

The Output Destination Editor dialog appears.

If you modify an existing destination, keep in mind that the destination changes for all jobs that use this destination.

3. If you are creating a new destination, enter a unique name for the output to make it easy to distinguish it from other destinations listed on the Output tab.
4. Mark **A folder or network share** and specify that job's destination folder in the Technical Publications directory as follows:

Job	Folder
Application Gateway	\PKG STUDIO\Application Gateway\App Gateway Help
Management Reports	\PKG STUDIO\Management Reports\Mgt Reports Help
QA	\PKG STUDIO\QA Help
WB Inline	\PKG STUDIO\Pkg Studio Inline Help\HTML
WfWI	\WfWI\WfWI Help
WfWI Debugger	
WIE Debugger	\PKG STUDIO\Pkg Studio Help
WiseScript Express	
WPS	

5. Click OK on the Output Editor Destination dialog.
6. On the Output tab, mark the checkbox of the destination you just created or edited.

Updating AutoMap Jobs

Before running an AutoMap job for a new release, use the following checklist to make sure the job's settings and templates are up-to-date.

What to check	What to do
The names and order of the FrameMaker files.	<p>For each project in a job, verify that the name and order of the FrameMaker files match what is in the FrameMaker book. For details, see Inputs Tab on page 205.</p> <ul style="list-style-type: none"> ◆ If the name of a file has changed, delete it and add the file with the new name. ◆ If there is a new file, add it. ◆ If a file in a FrameMaker book has been replaced by a file that is imported from another source (example: a file that is single-sourced), delete it and add the new file. ◆ Make sure the files are in the same order as they are in the FrameMaker book, except for the Front Page.fm file.
Conditional Text	<p>If any project in a job has new or updated conditions in the FrameMaker files, add or update the job's conditions. Also, add these conditions to the tables in this document that list the conditions for each job. Because you can only apply conditions to a job and not to an individual project, make sure the conditions for one project do not conflict with those of another. (Example: You cannot include the WiseScript Express project in the WPS job, because this would require the Express condition to be both visible and hidden.) For details, see Conditions Tab on page 207.</p>
Variables	<p>If any project in a job has new or modified variables, add or update the job's variables. Also, add these variables to the tables in this document that list the variables for each job. For details see, Variables Tab on page 208.</p>
HelpBuild.ini	<p>Change the name of HelpBuild_WPS.ini or HelpBuild_WfWI.ini to HelpBuild.ini. See Importing Help Files into Visual SourceSafe on page 215.</p> <p>Any time a help file is added to or deleted from Visual SourceSafe, make the corresponding changes to the HelpBuild.ini file. If the name of a help file changes, change its name in the HelpBuild.ini file. When the VSS Help Automation tasks run, any help file that has changed is updated in Visual Source Safe using the HelpBuild.ini file. When you add or delete files or projects to the HelpBuild.ini file, be sure to update the appropriate num=N equations. See Importing Help Files into Visual SourceSafe on page 215.</p>
Scheduling	<p>Verify that all the jobs and the VSS Help Automation tasks are appropriately scheduled. See Scheduling AutoMap Jobs on page 211.</p>
Altiris WebWorks Template	<p>If you are creating help that uses the Altiris WebWorks templates, make sure you are using the most current template. Usually, you should only have to update the .WPF file. There are two Altiris WebWorks templates (one for CHM help and one for Web help) at \\Docdev\Docs\Doc_Standards\FramerMaker\WebWorks_Templates, which we store locally at V:\Technical Publications\Help\WebWorks Templates.</p> <p>If you update the Altiris WebWorks template for creating CHM help, do the following to modify where it looks for the hhc.exe program.</p> <ul style="list-style-type: none"> ◆ Double-click the template to open it in WebWorks Publisher. ◆ Select Project menu > Options. ◆ Click the HTML Help tab. ◆ In the Install Location field, enter the correct path to the hhc.exe program. Example: C:\Program Files\HTML Help Workshop.
FrameMaker tags and markers	<p>If you have added tags or markers to the FrameMaker template (example: a new paragraph tag), then these tags and markers must be added to the WebWorks template and mapped to the appropriate WebWorks style. See Adding New Mappings to Project Mappings on page 189.</p>

Scheduling AutoMap Jobs

Each AutoMap job can be scheduled to run automatically. Because the jobs need to run when the Frame files are not being accessed, we normally schedule the jobs to run at night. The scheduling of the jobs has to consider the following:

- ◆ The jobs must run before the tasks in the Scheduled Tasks folder check them into VSS; see [Importing Help Files into Visual SourceSafe](#) on page 215.
- ◆ The jobs must not overlap, especially if they use the same Frame files.
- ◆ The WfWI job currently takes about 35 minutes and WPS takes about 1 hour.

To schedule jobs in Automap:

1. Double-click a job.
The Job Properties dialog appears.
2. Click the Basic tab and click Schedule.
The scheduling dialog appears.
3. Click the Schedule tab and specify when the job should run.
4. Click the Task tab and mark **Enabled**.
If you don't mark **Enabled**, the Job does not run as scheduled.
5. Click OK to close the scheduling dialog.
6. Click OK on the Job Properties dialog.
7. Check the Next Run column in the AutoMap window to verify that the schedule you set for the job is correct.

To schedule jobs from the Scheduled Tasks folder.

1. Select Scheduled Tasks from the Control Panel.
The Scheduled Tasks dialog appears.
2. Double-click the job you want to schedule.
3. Click the Schedule tab and specify when the job should run.
4. Click the Task tab and mark **Enabled**.
5. Click OK to close the dialog.

Using AutoMap to Create PDFs

Though we have licenses for creating PDFs with AutoMap, we currently (11/19/03) don't use them for the following reasons:

- ◆ Before creating a .PDF, we would have to manually set up the TOC and index files.
- ◆ After creating a .PDF, we would have to manually replace the first page with the cover art.

If we can automate these steps, then creating PDFs with AutoMap would be a significant time saver.

AutoMap also doesn't create .PDFs with working X-book cross-references (although their help says it does). If WebWorks fixes this issue, we should use AutoMap to create .PDFs because our current procedure also doesn't create working X-book cross-references.

We could also use this option to automatically update the .PDF of our DocProcedures on a regular basis.

Troubleshooting AutoMap and Publisher

When AutoMap fails to run a job successfully, a warning dialog usually appears. You can also check the status of a job's last run in AutoMap's Last Run Status column. For a list of the possible statuses, see [The WebWorks AutoMap Window](#) on page 202.

See:

[Checking a Job's Log](#) on page 212

[Fixing Broken Cross-References Using a Job's Log](#) on page 212


[Accessing WebWorks Support](#) on page 213

[Known WebWorks Problems](#) on page 213

[Working Around the hhc.exe AutoMap Error](#) on page 214

Checking a Job's Log

Each time AutoMap runs a job, it creates a log that lists everything that happened when the job ran. Use this log to determine what went wrong when a job fails and to locate broken cross-references.

1. In AutoMap, select the job and click .

The directory that contains the log for that job opens.

You can also open a job's log without using AutoMap. In the V:\Technical Publications\Help\WebWorks Jobs directory, there is a directory for each AutoMap job. In each of these directories, there is a logs directory that contains a log for each time AutoMap ran that job.
2. Locate and open the most current log.
3. Search or scan the log for errors or warnings.

If a warning concerns a problem with a Frame file, fix the problem in the Frame file. If you get an error message that you cannot resolve, report the problem to WebWorks. See below.
4. To search for warnings for broken cross references only, search for "X-ref."

The log lists broken cross-references that are often missed by other means of testing the help. Example: If a X-book reference is broken, it does not appear as a hyperlink in the help but only as italic text. This is easily missed when checking hyperlinks.

Fixing Broken Cross-References Using a Job's Log

1. Open the latest log for the job. See above.
2. To find each broken cross reference, search for "x-ref."
3. If the job has more than one project, scroll up from the broken cross-reference warning to locate the first occurrence of "Generating all for project file." This line of the log identifies the project that contains the broken cross-reference.
4. In the warning, note the file that contains the unresolved cross-reference, and the file to which the cross-reference points.
5. Note the topic that immediately precedes the warning, this is the topic that contains the unresolved cross-reference.
6. Open the .CHM file for the project, locate the topic that contains the broken cross-reference, and verify that it is broken.
7. Locate and fix the broken cross-reference in the FrameMaker file.

Accessing WebWorks Support

To report problems to WebWorks:

1. Access the WebWorks support area by going to <http://www.webworks.com/support>.
2. Click the Support Request link.
If this is the first time you have accessed the Support Request page, you will have to log in using JoAnn's e-mail address and her WebWork's password.
3. On the Support Request page, supply all the information requested, including attaching appropriate files, and submit the request.

To check the status of problems submitted to WebWorks:

1. When WebWorks needs more information, resolves the issue you submitted, or responds to the issue in some other way, they send JoAnn an e-mail. After she forwards you this e-mail, open it and click on Your WebWorks Support History.
This takes you directly to My Support Area that lists all of the issues that we have submitted.
2. Find and click the appropriate issue.
This opens a Detailed History for that case number.
3. Scroll down to a table that contains a table for Comments.
These comments contain WebWork's response to the issue as well as any comments or actions you have taken since submitting the Request.

You can also go directly to My Support Area using the following URL: <http://www.webworks.com/support/mysupporthistory.aspx>. This only works if you have previously submitted a request.

To resolve license problems:

If you encounter license problems, WebWorks might ask for your license.log and license.dat files. These are located in the C:\Program Files\Quadralay\WebWorks License Manager 3.0\etc directory. For information on adding AutoMap license keys, see [Installing WebWorks AutoMap](#) on page 200.

Known WebWorks Problems

The following known issues have been reported to WebWorks:

◆ hhc.exe errors

See case # 4177. We randomly encounter hhc.exe errors. This causes the .CHM file associated with this error message to be corrupt. WebWorks has not been able to reproduce and resolve this problem. Currently, it only happens with the WfWI help. It may be resolved with build 8.0.6.2138. See [Working Around the hhc.exe AutoMap Error](#) on page 214.

◆ Parentheses in topic headings in Web Help

See case # 3206. When creating help for Web applications, you cannot use parentheses in topic headings. A parenthesis causes the navigation buttons to be grayed out. If you change the value for Pages on the Project Options dialog to a value other than \$H;, you can then use parentheses, but then the topics are not named properly. <<<After the 8.0.6.2138 build has been installed, this should be tested.>>>

◆ **Duplicate files created for .TIF graphics in Web Help**

If you reference the same graphic more than once in the Frame files, use a .JPG instead of a .TIF. Each time you reference a .TIF, a separate .GIF is created in the help files.

◆ **Internal Frame Error**

See case # 4258. We also encountered an internal FrameMaker error during the WfVS job. This job no longer exists, but the workaround might be helpful if the same error occurs again. The error was resolved by opening the Frame book and setting the appropriate conditions for the help being created. If this happens again, reopen this case.

Working Around the hhc.exe AutoMap Error

Sometimes AutoMap encounters an hhc.exe error when it uses HTML Help Workshop to compile the HTML files into the .CHM file. This error occurs when the HTML Help Workshop is unable to execute the hhc.exe program. You can work around this problem by manually running the hhc.exe program to compile the .CHM.

1. In the V:\Technical Publications\Help\WebWorks Staging Area\ directory, locate the job that failed and open its Output directory.
There is a subdirectory in the Output directory for each project in the job.
2. Open the Project directory where the error occurred.
This directory contains all the HTML files that were created, a .CHM file that does not work, an .HHP file, and several other files.
3. Double-click the .HHP file.
The HTML Help Workshop opens.
4. Select File menu > Compile.
The Create a compiled file dialog appears.
5. Click Browse and select the .HHP file again.
This step is necessary because of the HTML Help Workshop's inability to correctly resolve file paths.
6. Click Compile on the Create a compiled file dialog.
The .CHM file is compiled and replaces the defunct .CHM file.
7. Copy the file to one of the following locations:
 - Its output directory as specified in its AutoMap job. See [Output Tab](#) on page 209. The next time the VSS Automation Help tasks run, the file will be imported into Visual Source Safe.
 - Visual Source Safe. Use this option if the VSS Automation Help tasks are not set to run.

Importing Help Files into Visual SourceSafe

After WebWorks AutoMap generates our help files, a program written by Mike Rudge (VSS_Help_Automation.EXE) automatically imports them into VSS. An .INI file (HelpBuild.ini) lists all of these files with their location in VSS, Tech Pubs, and the WebWorks staging area. There are also two tasks in the Control Panel's Scheduled Task folder that control when this program checks files in and out of VSS.

VSS Help Automation

This program checks help files out of VSS to a staging area, copies the new help files to this staging area, and then checks these files back into VSS. These transactions take place automatically according to the schedules of two tasks in the Scheduled Tasks folder (currently on Seth's computer). The VSS Help Automation program can be modified by editing the VSS_Help_Automation.wse file. This program and its accompanying WSE file are at V:\Technical Publications\Help\WebWorks License Keys and Program Files\Automation.

HelpBuild INI File

When VSS Help Automation runs, it uses a HelpBuild.ini file to know which files to check in and out of VSS. In the V:\Technical Publications\Help\WebWorks License Keys and Program Files\Automation directory, there are two .INI files: HelpBuild_WPS.ini and HelpBuild_WfWI.ini. We could combine these two files into one HelpBuild.ini, which would eliminate the need to rename these files. However, when creating help for WfWI, hundreds of WPS help files would then be checked in and out of VSS each night. Because we have 2 different .INI files, you must change the name of HelpBuild_WPS.ini or HelpBuild_WfWI.ini to HelpBuild.ini before running VSS Help Automation.

After each release, archive the .INI file for that release in V:\Technical Publications\Help\WebWorks License Keys and Program Files\Automation\INI Archive.

The HelpBuild.ini file has three sections:

- ◆ The first section specifies the GlobalPaths, which is currently VSSWorkingDir=C:\Build3. This is the staging area for the files.
- ◆ The second section contains a list of all of the projects and begins with an equation giving the number of the projects (example: Num=49). Each project is listed using the equation: Project<n>=<name>, where n is the number of the project and name is the name of the project (example: Project1=WPS_CHM).
For HTML help, each AutoMap job has one project. For Web help, each directory of the output constitutes a project. Example: Application Gateway has 21 projects because it has output in 21 different directories.
- ◆ The third section provides information about the files for the projects listed in section two. This consists of:
 - The name of the project.
 - The location of the project's files in VSS.
 - The location of the project's files in the staging area.
 - The location of the project's files in the Tech Pubs directory.
 - The number of files in the project, for example Num=6.
 - A list of all the project's files using the formula file<n>=<name>, where n is the number of the file and name is the files name.

Scheduled Tasks

The Scheduled Tasks folder in the Control Panel (currently on Seth's computer) has two tasks that determine when help files are checked in and out of VSS:

- ◆ VSS_Help_Automation_CheckOut
- ◆ VSS_Help_Automation_CheckIn

The CheckOut task must run before AutoMap runs <<<need to verify. SS 11/4/04>>>. The CheckIn task must run after AutoMap runs; see [Scheduling AutoMap Jobs](#) on page 211.

If these tasks need to be re-created, use the following procedure:

1. Double-click the Scheduled Tasks folder in the Control Panel.
2. Double-click Add Scheduled Task to run the Scheduled Task Wizard.
3. When asked to select the program that you want Windows to run, specify VSS_Help_Automation.EXE and click Next.
It is at V:\Technical Publications\Help\WebWorks License Keys and Program Files\Automation.
4. Change the name of the task to VSS_Help_Automation_CheckIn.
5. Mark when to perform the task and click Next.
You can easily change this later.
6. Select the required specifics for the tasks schedule and click Next.
7. Enter your network password, click Next, and click Finish.
8. Open the task and modify the command in **Run** by adding the argument /checkin.
9. Click OK.
This creates the VSS_Help_Automation_CheckIn scheduled task.
10. To create the VSS_Help_Automation_CheckOut scheduled task repeat the above steps, but use CheckOut in the task's name and modify the command in **Run** by adding the argument /checkout.

Troubleshooting

There are several ways to verify that the files generated by AutoMap are successfully imported into VSS. You can check VSS to make sure it has the latest file. You can also check the logs that are created as files are checked in and out of VSS.

There are 3 logs for each project listed in the HelpBuild.ini file (VSS_CheckIn.log, VSS_CheckOut.log, and VSS_UndoCheckOut.log). These logs are saved in the staging area (C:\Build3) in the same directories as the help files for that project. The structure of the directories in the staging area parallels the structure of the directories in VSS that contain help files. If files are not copied correctly, these logs record these problems.

Adding Bread Crumbs to a Help File

As of July 2004, we do not use bread crumbs. However, we might consider using them in the future. Seth gathered the following information from the class he took on Using WebWorks Functions.

To add bread crumbs to a help system, you need to make changes to the template in 4 areas:

1. Create the following 3 custom building blocks:

- **BC80Breadcrumbs1**

```
$COMMENT(
```

This BB is used to create bread crumbs at the top of each output page.

It is referred by Title para.

```
);\
```

```
@SET_GLOBAL(BreadCrumbs, $DATA(raw));\
```

```
@SET_GLOBAL(BreadCrumbs1, <a href="$PAGE(html,name);"> $DATA(raw);
```

```
</a>);
```

- **BC80Breadcrumbs2**

```
$COMMENT(
```

This BB is used to create bread crumbs at the top of each output page.

It is referred by Chapter para.

```
);\
```

```
@SET_GLOBAL(BreadCrumbs, $GET_GLOBAL(BreadCrumbs1); | $DATA(raw));\
```

```
@SET_GLOBAL(BreadCrumbs2, $GET_GLOBAL(BreadCrumbs1); | <a
```

```
href="$PAGE(html, name);">$DATA(raw); </a>);
```

- **BC80Breadcrumbs3**

```
$COMMENT(
```

This BB is used to create bread crumbs at the top of each output page.

It is referred by Heading1Split para.

```
);\
```

```
@SET_GLOBAL(BreadCrumbs, $GET_GLOBAL(BreadCrumbs2); | $DATA(raw));\
```

```
@SET_GLOBAL(BreadCrumbs3, $GET_GLOBAL(BreadCrumbs2); | <a
```

```
href="$PAGE(html, name);">$DATA(raw); </a>);
```

2. Reference these building blocks at the end of 3 paragraph style macros:

Paragraph Style	Building Block to Add
ChapterTitle	\$BCBreadcrumbs1
Heading1	\$BCBreadcrumbs2
Heading2	\$BCBreadcrumbs3

3. Add the following line of code to the Normal.asp file:

```
<p class="BREADCRUMBS">$UMYouareHere; $GET_GLOBAL(BreadCrumbs);</p>
```

Place the code before the code that begins <!--BeginHiddenExpansion \$BP80PageASP_Content; EndHiddenExpansion-->

You change the display of the bread crumbs by inserting the appropriate HTML code in the bread crumbs script in the Normal.asp file.

4. Add a User Macro called UMyouareHere:

The content of this macro is the text that appears before the bread crumbs. WebWorks put "You are here:" in this macro.

Chapter 13

Testing and Troubleshooting

Topics include:

- ◆ [Testing Help.](#)
- ◆ [Final Testing of Documentation Deliverables.](#)
- ◆ [Troubleshooting.](#)
- ◆ [Enabling All Options in Package Distribution.](#)

Testing Help

Print this help plan and provide it to anyone who is testing online help. These guidelines apply to all Wise products.

Help Deliverables to Test in Each Product

Product	What to test
Wise for Windows Installer	WfWI (Wise editor) Visual Studio Integrated Editor Setup Editor inline help (lower right pane) Debugger Software Manager WiseScript Express
Wise Package Studio	Workbench Workbench inline help (right pane) Application Gateway ConflictManager Software Manager Windows Installer Editor WIE / Setup Editor help (from lower right pane) WIE / Debugger WiseScript Editor WiseScript Express (Standard Edition only)

What to Test in Each Help Deliverable



Details for each step follow.

Fonts
Hyperlinks
Context-sensitive help
Inline help (WfWI and WIE Setup Editor, Workbench)
Table of contents
Index
Search
Favorites

Testing Fonts in Help

1. Install the help on every operating system we support. You do not have to install the entire product; just copy the .CHM files to the computer and double-click them to open.
2. Scroll through a few pages and make sure the fonts look OK.
 - First, is the text readable?
 - Second, are the bullet characters correct? If bullet characters look different from any of the ones shown below, there is a font problem. Report it.
 - ◆ First level bullet (usually shows up as "t" if there's a problem)
 - Second level bullet
 - Third level bullet (usually shows up as "Ø" if there's a problem)

Testing Hyperlinks in Help

1. Select Help > Help Topics. The help window should open and display the Welcome topic.
2. Click every hyperlink on the first topic, using the Back tool () at the left of the toolbar (or press Alt+left arrow) to return to the original topic each time.
 - Make sure the link works, make sure it goes to the right topic.
 - Make sure all Internet links go to the right web address.
 - Make sure all SDK links open the correct topic in the Windows Installer SDK Help.
3. When you've tested all the links on the first topic, go to the next page and repeat. To go to the next page, click the Next button () at the bottom of the topic page.

Testing Context-Sensitive Help

From EVERY window and dialog in the product, press F1. Make sure a topic opens. Make sure the topic is appropriate to the window or dialog.

Additional guidelines and tips:

- ◆ When you press F1 in the Tables tab in Setup Editor (in WfWI, VS integrated editor, WIE), the following behavior is expected:
 - If no table is active, display the top level topic about tables from the Wise help.
 - If a table is active, display the appropriate SDK help topic for that table. If the SDK topic cannot be found, display the top level topic about tables from the Wise help.
- ◆ The F1 key doesn't work in the Download Redistributables Wizard. Instead, click the Help button on each dialog in the wizard.
- ◆ In Wise Package Studio, there are additional guidelines for testing tools. For each tool that accepts either a .WSE or .MSI/.WSI, you must test the entire tool with each type of file. Example: run ApplicationWatch. It asks you to specify a file. Specify a .WSE file and then run through the entire tool, pressing F1. Then run ApplicationWatch again, specifying a .MSI/.WSI. Run through the entire tool, pressing F1 in each dialog. This is because, depending on the initial file specified, a different .EXE (WfWI.exe or Wise.EXE) is accessed and processes help.
- ◆ In WfWI and WIE, create an installation that contains at least 2 releases. Some dialogs appear only when there are multiple release. Example: When you click the Test, Run, or Debug buttons, if there are multiple releases, an additional dialog appears asking which release to use. Press F1 on this dialog.
- ◆ In the Visual Studio integrated editor (formerly WfVS), Project Settings dialogs are accessible from two places; test both.
 - Create a new installation project and select Setup Wizard in the New Project dialog.
 - Create or open an installation project. Right-click on the project in Solution Explorer and select Properties.

Displaying Hard-to-Find Dialogs

- ◆ **Wildcard Deleted Files dialog (WfWI, WIE)**
 - On the Files page, pick folder and click Add Contents.

- In the Add Contents dialog that appears, mark the **Update...** checkbox.
 - Save the installation.
 - On your hard disk, remove a file that is in the folder that you clicked Add Contents for.
 - Save again.
 - You should now get the Wildcard Deleted Files dialog.
 - Press F1, and you should now get the Help topic that deals with the “Add Contents” dialog.
- ◆ **Files in Merge Modules dialog (WfWI, WIE)**
In Installation Expert > Files page, in the lower left list box, select the System32 directory, then add Winnt\system32\mfc42.dll.
- ◆ **Download Redistributables (WfWI, WIE)**
In Installation Expert > Files page, in the lower left list box, select the System32 directory, then add Winnt\system32\mfc42.dll. First the Files in Merge Modules dialog appears, then the Download Redistributables wizard.

Testing Inline Help

In Setup Editor (WfWI, WIE only)

1. In WfWI and WIE, go to Setup Editor and display every inline help page (in the lower right pane).

Note

For a list of all the HTML help pages and how to access them, see the document <V:\Technical Publications\Admin\Notes Memos + Misc\WfWI Inline Help.xls>.

2. On each page, click every help link. Make sure the ones named “Online Help” open the correct Wise help topic; make sure the SDK links open the correct Windows Installer SDK topic.

In Wise Package Studio’s Workbench:

1. In Workbench, go to the Tools tab and click every tool, one at a time, to display its inline help in the right pane.
2. On each help page, click every help link and make sure they open the correct Wise help topic; make sure any SDK links open the correct topic in the Windows Installer SDK Help.
3. Create a project without a process and open it in the Projects tab. Click every tool, one at a time, to display its help. Check the links as described in step 2.
4. Create a separate project with *each* one of the predefined processes. For each project, open the project and click each task, one at a time, to display the task help. Check the links as described in step 2.

Testing the Help Table of Contents

1. Select Help > Contents.
2. Expand all books and make sure there are no “empty” books, that is, every book should have a topic name.

3. Click every link in the table of contents and make sure the correct topic opens. (This step can be skipped if testing time is limited.)
4. Repeat for all other helps in the product. Examples:
 - Software Manager
 - WiseScript Express
 - Standalone Debugger help

Testing the Help Index

It is impractical to test all the index entries, but you should spot-check the index to make sure it works.

1. In the Help window, click the Index tab.
2. Double-click any index entry. The help topic should open at the place where the index entry points to; this might not be at the top of the topic.
3. Scroll through the index and repeat for additional index entries at random.

Note

Typically, when an index entry has sub-entries, it is not linked to a topic, therefore when you double-click on it a new topic will not appear. This is not a bug.

To test the help search:

1. In the Help window, click the Search tab.
2. In **Type in the keyword to find**, type a word or phrase that you think is likely to be in this particular help file and click the List Topics button.

One or more topics should appear in the **Select Topic to display** list.

To test the help Favorites:

1. In the Help window, display any topic.
2. Click the Favorites tab.
3. Click the Add button at the bottom of the Favorites tab.
The topic should be added to the Topics list.
4. Click the Contents tab and display a different topic.
5. Follow steps 1 through 4 above to add several more topics to the Favorites list.
6. Click the Favorites tab and double-click the topics listed there.

The appropriate topics should open.

Reporting Help Problems

As you encounter problems with help, keep a Word or Notepad document and send it to the help developer when you finish testing. Here's what you need to record:

- ◆ If the F1 help doesn't work, record what dialog or window you were on. If it's one that's hard to get to, include a hint on how to get there.
- ◆ If the F1 help displays the wrong topic, record what dialog you were on and the title of the topic that appeared.
- ◆ If the problem is with a link in a help topic, record the topic title, which is the heading displayed at the top of the topic window.

Suggestions for Dividing the WfWI Help Testing

Split the WfWI help testing by having one person take each of the following tasks.

- ◆ Test all the hyperlinks. You can split this up; just have each person take a different group of books in the table of contents.
- ◆ Test context-sensitive help in Installation Expert.
- ◆ Test context-sensitive help in MSI Script.
- ◆ Test context-sensitive help in Setup Editor.
- ◆ Test context-sensitive help in WiseScript Express. Select Tools > WiseScript. Make sure WiseScript Express opens and not WiseScript Editor.
- ◆ Test context-sensitive help in the tools and misc. menu items:
 - Everything in the Tools menu.
 - Test, Run, Start, and Debug buttons.
 - Download Redistributables Wizard on the Help menu. Remember that F1 doesn't work in this wizard; you must click the Help button on each dialog.
- ◆ Spot-test help in the Visual Studio Integrated Editor:
 - Test a few hyperlinks.
 - Test an SDK link to make sure the SDK help opens.
 - Test context-sensitive help in areas that are unique to the Visual Studio integrated editor.
 - Setup Wizard from the File > New Project menu.
 - Project Settings dialogs.
 - Start Without Debugging on the Debug menu.

Final Testing of Documentation Deliverables

After all documentation deliverables are checked in and the next build is available, verify the documentation files in the product. To do so, in each executable (Workbench, Software Manager, Debugger, and so on), select every documentation-related option from the Help menu and check the following.

For a list of all the deliverables in a particular release, you can print the deliverables spreadsheet. Typically, this spreadsheet is in Technical Publications\Admin\Estimates + Schedules. If you cannot find it, ask the Technical Publications Manager.

.PDFs (References, Getting Started Guides)

- ◆ Does the correct book open?
- ◆ Does it have the correct cover? Look carefully at this; they are all very similar.
- ◆ Is the TOC collapsed?
- ◆ If the book is single sourced, is the correct variable used?

CHM Help

- ◆ Does the correct help open?
- ◆ Press F1 from one or two places in the UI. Does the correct help open?

Release Notes

- ◆ Does the correct document open?
- ◆ Is the release number correct in the title bar? At the top of the page? In the Contents hyperlinks and the headings?
- ◆ Do all hyperlinks in the document work?

Troubleshooting

Use this section to record resolutions to problems you encounter with the book, help, or .PDF.

See:

[Verifying Tutorial Files in the Product](#)
[Troubleshooting Help](#)
[Troubleshooting the Frame Book](#)
[Troubleshooting the .PDF](#)

Verifying Tutorial Files in the Product

The tutorials in our documentation depend on certain items being installed into the Wise application directories. Check these items:

- ◆ Is a tutorial or evaluator’s guide shipping? If so, is the appropriate tutorial folder present, and does it contain file names and pathnames that match what is written in the tutorial or evaluator’s guide?
- ◆ In all Tutorial folders, is there a readme.txt file that explains what the files are for, and where the tutorial text can be found? Is this readme.txt file still correct?

Troubleshooting Help

Context-Sensitive Help

The most common problems in the context-sensitive help are missing IDs and duplicate IDs. Development must fix missing IDs and IDs that are duplicated in the product. If duplicate IDs are caused by two topics mapped to the same ID, we can fix them by editing the Map ID file.

You find missing IDs when you press F1 in the product and an ID is not displayed. You can find duplicate IDs by compiling the help project and displaying the Duplicate Map IDs report (Tools > Reports > Duplicate Map IDs). To easily find duplicate IDs, copy the Duplicate Map IDs report to Notepad. Search for *Duplicate Map #s*. That takes you right to the duplicate IDs.

Troubleshooting the Frame Book

Chapter is Numbered 1

Sometimes a chapter in a book is numbered “1” when it isn’t the first chapter.

Simple Fix

1. Open the chapter and reset the numbering. See [Reset the Document Numbering](#) on page 155.
2. Open the chapter that was mis-numbered.
3. If it hasn’t been fixed, try this:

Copy the Bad Chapter to Another File

1. Open a chapter in the book whose numbering is OK.

2. Delete everything in the chapter EXCEPT the first line, which is the chapter number (with the ChapterNo tag).
3. Open the chapter that's mis-numbered.
4. Copy everything in the chapter EXCEPT the first line and paste it into the other document.
5. Close the mis-numbered document.
6. In the good document, do a Save As and name it the same as the mis-numbered document. Overwrite the existing document.
7. Update the book.

Open the chapter that was mis-numbered to verify that it's been fixed.

Also see the Adobe Knowledgebase article "Resolve autonumbering problems in FrameMaker" at www.adobe.com/support/techdocs/203d6.htm. If this link doesn't work, search by the document title or by number 324605.

Extra Pages Appear at the End of a Chapter

Often, hiding some of the conditional text in a document leaves blank pages at the end of the chapter. Sometimes Frame doesn't delete those extra pages. This can also happen when you delete several pages of text from a chapter.

1. Open the chapter and select Format > Page Layout > Pagination.
2. Mark **Single Sided** and select **Delete Empty Pages** from the drop-down list.
3. Click Set.
4. Close and then re-open the chapter.
5. If it hasn't been fixed, follow the steps in [Copy the Bad Chapter to Another File](#) on page 226.

Missing Fonts Error in Frame

This information is from Tim Murray of TechKnowledge Corp. (www.techknowledgecorp.com/help/index.html)

Problem

You open a Frame file and the console reports that a font was not found, so another was substituted. Additionally, you find that you cannot do a generate because Frame tells you it can't properly open all the files.

Load the font

The obvious fix.

"Don't care" solution

In this solution, you are positive you never used the font, but even if you did, the substitution FrameMaker provides would be okay anyway.

1. Close the file.
2. Turn off the "Remember Missing Font Names" preference.
3. Open the file (you will get the error again) and save.

The next time you open it, the message will be gone.

“Programmed” solution: single font

In this solution, your goal is to substitute the missing font for one that is different than the choice FrameMaker provides. In this example, let’s say that Frame wants to substitute Times, but you would really rather use New Century Schoolbook.

1. In preferences, disable Remember Missing Font Names.
2. Exit Frame. Do not save the document!
3. Edit the maker.ini line “DefaultFamily=”. In most cases it is Times New Roman, Times, Tms Rmn. Insert the font of choice at the front:

```
DefaultFamily=NewCenturySchlbk, Times, Times New Roman, Tms Rmn
```

You don’t have to have anything after your initial choice, but it doesn’t hurt. It simply tells Frame to try each one in this order.

4. Make sure your new font (New Century Schoolbook) is active.
5. Start Frame, and open the file. Console should report that your original font is missing and the new font was substituted.
6. Save the file. The change is complete.

“Programmed” solution: multiple fonts

The overall approach is similar to the single-font problem, except that the solution is made more complex by the need for numerous changes. In summary, you will utilize Frame’s substitution tables found later in the .INI file. It is beyond the scope of this page to train the reader in how to work with this file (actually, the comments in the maker.ini file are rather descriptive); however, I can offer these tips:

1. Read up on the Adobe Knowledge article www.adobe.com/support/techdocs/112da.htm.
2. When you have more than one A=B line covering the same font family, place the most detailed lines first, and leave the wildcards for later:

```
FontX, Italic, *, Regular = FontY, Oblique, *, Regular
```

```
FontX, *, *, * = FontY, *, *, Regular
```

In this example, the *, *, * catches those that don’t get caught by the first one.

3. It’s not required if your .INI is coded properly, but you can use the DefaultFamily setting (described above) to grab your most commonly used font.
4. Over the years I’ve developed (what I think anyway) is a pretty nifty font conversion section of the maker.ini file. It covers lots of common PostScript <> TrueType conversions, some Bitstream <> Adobe, and a few other items. You can get it at www.techknowledgecorp.com/public/font_ini.txt.

About books and generates

You may be in a situation where you need to do a generate, but in order to do so you would have to override the fonts . . . but you don’t want to because doing so would mess up the design. The solution is to open the file or files that are missing the fonts; since they are already open during the generate, you won’t get the “cannot open file” error message.

Troubleshooting the .PDF

Fonts are Jagged or Image Resolution is Bad

If, when you open the .PDF in Acrobat, the fonts look bad or the images are blurry, check the following:

- ◆ In the Frame book, did you select Adobe PDF in the Print Setup?
- ◆ Did you open Distiller before saving the book, and did you select the appropriate Distiller settings as described in [Setting Up Acrobat Distiller](#) on page 166?

Old RoboHelp Issues

The following issues are obsolete because we no longer use RoboHelp.

Fixes Don't Appear in Recompiled Help

If you make changes in a help project, recompile it, and then do not see the changes in the updated .CHM file, it's probably because the file didn't really get recompiled. This happens when you've checked the .CHM in to Visual SourceSafe, which sets the .CHM in your working folder to read-only. When you recompile the help, RoboHelp can't overwrite the read-only file. Unfortunately, it doesn't display any errors message; it just doesn't make the file. If you don't check the date/time stamp on the file, you are likely to check in the old copy of the file, which will not contain your fixes.

If your changes don't appear in recompiled help, check the date/time stamp to make sure the file got recompiled. If the file did not get recompiled, check it out of SourceSafe or clear the **Read-only** checkbox in the file's Properties dialog. Then compile again.

Page Numbers in Help Hyperlinks

This happens in the following cases:

- ◆ When you apply the Frame-to-HTML template to the Frame files. For some reason, if you apply the style template to a single file, or if you do not mark the checkboxes in the Import Formats dialog exactly as instructed, page numbers are left in cross-references that point to other chapters in the book. Cross-references to topics within the same chapter are not affected.

To prevent this, open the book, select all the chapters, and apply the template. Also, in the Import Formats dialog, select ONLY the checkboxes that are shown in the instructions.

- ◆ If cross-refs are linked to a chapter that is not in the book. Example: This happens if you copy a section from one book to another book, and that section included a x-ref, and the x-ref remained linked to the original book's file. The cross-ref does not appear broken until it gets into HTML help, and then it displays a page number in the link. This can also happen if a chapter is removed from a book, but the file remains in the book folder. The cross-links still go to the obsolete file.

To prevent this, make sure you get rid of obsolete files in book folders during Finishing Steps. See [Finishing Steps Checklist](#) on page 153.

Note

The second instance might still cause a problem with WebWorks. Test to see what happens. JK 2/2004.

Hyperlinks Go to a Duplicate Topic

When you link to a topic and the text is wrong but the topic title is right, your help project probably has duplicate topics. There are two instances in which the RoboHelp project might contain duplicate topic names:

- ◆ If the book contains two topics with the same name. Example: Wise for Windows Installer has a chapter named *Distributing Your Installation*. That chapter also had a

subtopic with the same name. Fix these instances when you find them and avoid creating them in the first place.

- ◆ If you merge books into one help topic, as in Wise for Visual Studio .NET. Example: The topic that describes the Remark action in MSI Script is named *Remark*. WiseScript Express also contains a Remark action, and its book contains a topic named *Remark*. Another example is the Welcome chapter; all books have one, so when you merge books into one help project, all the Welcome topics are duplicated. In such cases, you typically want to leave the topic names as is.

If RoboHelp finds a duplicate topic name during the file conversion, it appends a number to the end of the second file name. Example: The file for the first topic named *Distributing Your Installation* is named `Distributing_Your_Installation.htm`. If RoboHelp encounters another topic with the same name, it names the second file `Distributing_Your_Installation1.htm`.

RoboHelp does NOT rename duplicate files when the files are in different levels of the directory structure. This happens when you merge books. Example: All topics in the Wise for Visual Studio .NET help are one directory level down, as in `WfVS Help\Welcome\Welcome.htm`. Typically, the WiseScript Express book within the WfVS help is two levels down, as in `Help\WiseScript_Express\Welcome\Welcome.htm`. In this case, BOTH topics are named `Welcome.htm`.

How do you fix duplicate topics?

When RoboHelp renames duplicate topics, you just need to pick the correct file name when you create hyperlinks and map help IDs. Unfortunately, you might not even know a topic is duplicated until you test help and find that a hyperlink or F1 displays the wrong topic. Then, decide whether the link should go to the topic with or without the "1" in the file name, and make the appropriate changes in the Map ID file (.H).

If RoboHelp did not rename duplicate help topics, you need to rename one of them yourself.

1. In the RoboHelp Project tab, right-click the topic to rename and select Properties from the right-click menu.
2. Change the **File Name** field. Example: If the topic is named `Welcome.htm`, change it to `Welcome1.htm`.
3. Click OK.
RoboHelp renames the file and resets all links to that file.
4. Open the Map ID file and change the topic name there.
5. Regenerate the alias file (.ALI).

Enabling All Options in Package Distribution

The Package Distribution tool contains several options that are available only if you have specific third-party software installed. To enable all distribution options, Development has provided the following registry key:

1. Open Regedit.
2. Navigate to HKEY_CURRENT_USER\Software\Wise Solutions\Windows Installer Editor\Settings.
3. Right-click Settings and select New > DWORD value.
4. Name the value ForceShowDistribSystems.
5. In the right pane, double-click ForceShowDistribSystems. In **Value data**:
 - Enter 1 to activate this function.
 - Enter 0 to deactivate this function.
6. Click OK and close Regedit.

Shortcuts for Setting the Registry Key

The following .EXEs enable and disable this registry key:

- ◆ Distribution ON.EXE
- ◆ Distribution OFF.EXE
- ◆ Distribution Remove.EXE

These .EXEs are in Technical Publications\Admin\Documentation Software\Distribution Hack. Create shortcuts on your desktop if you plan to use them often.

Chapter 14

Department Procedures

<<<>>>.

Topics include:

- ◆ [Installing Product Builds.](#)
- ◆ [Reporting Time.](#)
- ◆ [Using Littlebuggy.](#)
- ◆ [FrameMaker Features You Need to Know.](#)

Installing Product Builds

Product builds are located in V:\Development\Builds; each product has its own subdirectory. (Example: V:\Development\Builds\Wise Package Studio.) The Builds directory for each product also contains a document named Serial Numbers.txt, which contains the serial numbers you can use to install the product. Because you probably will install a new build almost every day, you might want to make a shortcut to the Builds directory of the product you're working in.

Which edition should you install?

Usually, you should install the highest edition of the product. Install a different edition if you're writing about a feature that's specific to that edition, or to test how a feature works in different editions.

Do you have to uninstall a build before installing a new one?

Yes, unless you are specifically testing an upgrade installation, it's always best to uninstall the old build first. After you uninstall the build, always check the Program Files directory to make sure the application directory has been deleted. If not, delete it.

See:

[Tips for Installing and Uninstalling Wise for Windows Installer](#)

[Tips for Installing and Uninstalling Wise Package Studio](#)

Tips for Installing and Uninstalling Wise for Windows Installer

Every time you uninstall Wise for Windows Installer (WfWI), do the following:

1. Use Add/Remove Programs to uninstall the product.
2. Delete the share point directory, which is C:\Wise Share Point or C:\Wise Share Point Beta. The installation does not overwrite this directory, so if you have one left over from an old build it is left intact. Usually this isn't a problem, however, you can bet that the one time you forget to delete the share point will be the one time that a programmer has made a significant change there.
3. Look in the Program Files directory and, if the application directory is still there, delete it.
4. Delete the Wise registry key.

Note

Skip this step if you have other Wise products on your computer.

- From the Start menu, select Run.
 - In the Run dialog, enter regedit and click OK. The Registry Editor opens.
 - Expand HKEY_CURRENT_USER and expand the Software folder,.
 - Delete the Wise Solutions folder.
 - Close the Registry Editor.
5. If you use Software Manager, or if you're having problems with it, you might want to delete the data source periodically. You do not have to do this every time you uninstall.
 - From the Control Panel, select Administrative Tools and then select Data Sources (ODBC). The ODBC Data Source Administrator dialog appears.

- Click the System DSN tab.
- In **System Data Sources**, click **Software Manager Database** and click the Remove button.
- Click OK to close the dialog.

Tips for Installing and Uninstalling Wise Package Studio

The Wise Package Studio installation is more complex than for other products because of the databases it installs. In most dialogs in the installation, you can accept the defaults. In the Select Database Type dialog, do the following:

- ◆ Mark the **Microsoft Access** option. If you are documenting how a feature works with different databases, you might need to install a different database. In that case, ask for help.
- ◆ If you are specifically working in ConflictManager, you might want to mark the **Import sample applications into database** option, but it's not necessary.

Uninstalling Wise Package Studio:

Every time you uninstall Package Studio, do the following:

1. Use Add/Remove Programs to uninstall Wise Package Studio.
2. Delete the share point directory, which is C:\Wise Share Point or C:\Wise Share Point Beta. The installation does not overwrite this directory, so if you have one left over from an old build it is left intact. Usually this isn't a problem, however, you can bet that the one time you forget to delete the share point will be the one time that a programmer has made a significant change there.
3. Look in the Program Files directory and, if the Wise Package Studio Beta directory is still there, delete it.
4. Delete the Wise registry key.

Note

Skip this step if you have other Wise products on your computer.

- From the Start menu, select Run.
 - In the Run dialog, enter regedit and click OK. The Registry Editor opens.
 - Expand HKEY_CURRENT_USER and expand the Software folder.
 - Delete the Wise Solutions folder.
 - Close the Registry Editor.
5. If you use Software Manager or ConflictManager frequently, or if you're having problems with those products, you might want to delete the data sources periodically. You do not have to do this every time you uninstall.
 - From the Control Panel, select Administrative Tools and then select Data Sources (ODBC). The ODBC Data Source Administrator dialog appears.
 - Click the System DSN tab.
 - In **System Data Sources**, click **Software Manager Database** and click the Remove button.
 - Repeat the preceding step to remove the **Sample Wise Software Repository** and the **Workbench Database**. <<<get new names>>>
 - Click OK to close the dialog.

Reporting Time

All writers must record time spent on a specific release; this helps the Tech Pubs manager determine the accuracy of the estimate for that project and provides valuable information for producing future estimates. Contractors also need to report daily hours and submit an invoice to Wise for payment.

Recording Project Time

We use an Excel spreadsheet to record project-related time. Ask the Tech Pubs manager if you should use the master spreadsheet or a copy.

Time entry is simple; follow these guidelines:

- ◆ Enter your first initial in the **Who** column.
- ◆ Enter time in quarter-hour increments.
- ◆ Enter time as a decimal, such as 1.5 hours, rather than as hours:minutes.
- ◆ In the **Description** column, briefly describe what you did. This does not have to be detailed.
- ◆ At the top of the spreadsheet is a list of codes to enter in the **Area** column. This usually corresponds to the product and book you're working on. You can enter these codes in lowercase.
- ◆ At the top of the spreadsheet is a list of codes to enter in the **Activity** column. Use **wri** for all writing and editing as well as for all research and hand-off meetings with the programmers. You can enter these codes in lowercase.
- ◆ If a PRD number is available for the feature you are documenting, enter it in the **PRD#** column.

Note

You might find it more convenient to record your time on paper and periodically transfer it to the spreadsheet. Do whatever works best for you, but update the spreadsheet at least once a week.

What Not to Enter

Do not enter time that is not directly related to producing documentation for the release project. Examples: Training activities, department meetings, and invoice preparation are not directly related to the project.

Using Littlebuggy

Littlebuggy, our testing management tool, is the repository for all product defects. You use it to report defects and to get information about the defects you document. This section covers the Littlebuggy functions you are most likely to use; it is not meant to be comprehensive. For details on using other Littlebuggy features, see the Littlebuggy Help.



Accessing Littlebuggy

1. In your browser, go to <http://littlebuggy.altiris.com/aexhd/worker/>.
2. Log on using your altiris domain information, except precede your user name with altiris\. Example: altiris\my.name.

Littlebuggy Tips

- ◆ To sort the query display, click a column heading.
- ◆ When you open an item and then close it, the display is not resorted — the item you just opened is now somewhere else in the list. To prevent this, open items by right-clicking and selecting View in new window. But this just lets you view. To edit, click the pencil icon in the upper right. When you close the new window, the item is in its original place in the query list.

E-mailing From Littlebuggy

1. Display the incident to send.
2. Click the  icon to enter edit mode.
3. Click the  icon in the toolbar at the top of the incident.
4. Under "E-mail a message to specified recipients," enter one or more e-mail addresses or mark checkboxes for specific workers.
5. Click Done.

The e-mail is sent when you save the incident.

Also see:

[Creating an Incident in Littlebuggy](#) on page 237

[Creating Littlebuggy Queries](#) on page 239

[Common Littlebuggy Queries for Tech Writers](#) on page 239

[Adding Columns to the Littlebuggy Query Table](#) on page 240

Creating an Incident in Littlebuggy

If a field is not highlighted in the following screen shot, accept the default. Details on some of the fields are below the screen shot.

The following graphic contains a GraphicScale marker in an empty text frame.

New Incident

Time spent: 1 minutes | Total time: 0 minutes

Category: **Select one** | **Select one** | **Select if applicable**

Priority: **Select one** | Assigned: **See procedures**

Status: Open | Status 2: [current] 0 | Target Release: **Enter the release no.**

Title: **Enter a brief summary**

Steps to duplicate: **B I U** **Enter if necessary**

Comment: **B I U** **Describe the problem**

Comment visible to guests

Found in ver. build: **Enter version no.** | Fixed in ver. build: | Type: Bug

Explanation of change: **B I U** | Steps to test: **B I U**

Files affected: **B I U** | Area affected: **B I U**

Software change notice: *(release notes)* | Req'd: | *Mark if applicable* | Comp'd:
 Documentation: | *See procedures* |
 Documentation:
 Knowledgebase: |

Contact: --[none]-- | Find

Asset: --[none]-- | Find

External ID: | External Reference: | Schedule: Not scheduled

Start: 12/31/2004 | Due: 12/31/2004 | Estimated hours: **See procedures**

Percent complete: 0 | Actual hours: 0

Notify rules: Acknowledge contact
 Corp SE Notification <----- *Clear this*
 Incident reassigned

Attachments:

Linked to: | Linked by:

How to Complete Littlebuggy Fields

◆ Assigned

If	Assign it to
the incident is related to a specific new feature	specific developer
the incident is not related to a new feature	QA queue [<product name> QA]
the incident is strictly doc-related	Doc queue or specific writer

◆ Software Change Notice

This indicates whether the incident should be documented in the release notes.

If	Then mark:
The incident should not be added to the release notes	Req'd: <input type="checkbox"/> Comp'd: <input checked="" type="checkbox"/>
The incident should be added to the release notes	Req'd: <input checked="" type="checkbox"/> Comp'd: <input type="checkbox"/>
The incident has been added to the release notes	Req'd: <input checked="" type="checkbox"/> Comp'd: <input checked="" type="checkbox"/>

◆ Documentation

This indicates whether the incident should be documented in the book/help.

If	Then mark:
The incident should not be documented	Req'd: <input type="checkbox"/> Comp'd: <input checked="" type="checkbox"/>
The incident should be documented	Req'd: <input checked="" type="checkbox"/> Comp'd: <input type="checkbox"/>
The incident has been documented	Req'd: <input checked="" type="checkbox"/> Comp'd: <input checked="" type="checkbox"/>

◆ **Estimated Hours**


This indicates which writer is assigned to the item. Currently, Littlebuggy has no place to enter this information, so we use this otherwise unused field.

- 0 = None
- 1 = Teresa
- 2 = JoAnn
- 3 = Seth

If you are creating an incident for something that you know you are responsible for, then assign it to yourself when you create it. This will save your manager's time later.

Creating Littlebuggy Queries

Littlebuggy queries are written in SQL. If you are not familiar with SQL, copy the sample expressions in [Common Littlebuggy Queries for Tech Writers](#), or ask someone to write a query for you. The Program Manager, QA manager, and the developers can write SQL queries.

1. Open Littlebuggy and display the List Queries window (under Common Tasks, select Incident Queries).
 2. Click the  icon to create a new query.
The New Query window appears.
 3. In **Name**, enter a name for the new query.
 4. If the new query is based on an existing query, select the existing query from **Based on** and click Get Expression. The SQL expression appears.
 5. In **Expression**:
 - If the new query is based on an existing query, edit the existing expression.
 - If you are creating a new query from scratch, enter or paste the query expression.
 6. To test the new query, click Run.
 7. If the new query runs successfully, click Save.
- The new query is added to your personal query list.

Common Littlebuggy Queries for Tech Writers

The sample queries in this section are the ones most commonly used by technical writers. Copy and paste the sample expression into the **Expression** field on the New Query window and edit it as needed. See [Creating Littlebuggy Queries](#).

Doc-specific Incidents That Are Assigned to You

For CATEGORY_NAME, substitute the actual product name.

```
lower(workitem_category_tree_value) like lower('%' + 'CATEGORY_NAME' +
'%') and workitem_assigned_to_worker_id = $current_worker_id$ and
workitem_status_lookup_id = 300
```

Incidents Needing Documentation That Are Assigned to You

- ◆ For CATEGORY_NAME, substitute the actual product name. This appears twice in the expression.
- ◆ Find workitem_estimated_hours = n (near the end of the expression) and replace n with your assignment number.

```
lower(workitem_category_tree_value) like lower('%' + 'CATEGORY_NAME' +
'%') and workitem_require_documentation = 1 and
workitem_complete_documentation = 0 and workitem_estimated_hours = 1 and
workitem_status_lookup_id = 400 or lower(workitem_category_tree_value) like
lower('%' + 'CATEGORY_NAME' + '%') and workitem_require_documentation =
1 and workitem_complete_documentation = 0 and workitem_estimated_hours =
n and workitem_status_lookup_id = 600
```


Incidents Needing Documentation That Are Unassigned

For CATEGORY_NAME, substitute the actual product name. This appears twice in the expression.

```
lower(workitem_category_tree_value) like lower('%' + 'CATEGORY_NAME' +
'%') and workitem_require_documentation = 1 and
workitem_complete_documentation = 0 and workitem_estimated_hours = 0 and
workitem_status_lookup_id = 400 or lower(workitem_category_tree_value) like
lower('%' + 'CATEGORY_NAME' + '%') and workitem_require_documentation =
1 and workitem_complete_documentation = 0 and workitem_estimated_hours =
0 and workitem_status_lookup_id = 600
```

Adding Columns to the Littlebuggy Query Table

You might want to add documentation-related or other columns to the Littlebuggy query table.

1. Open Littlebuggy and display the Worker Report window (under Commands, select Incidents > Worker Report).
2. Click the  icon to format the query table.
3. From **Field**, select a field to add and click OK. Example: "workitem_require_documentation" or "workitem_complete_documentation."

Note

The field names are not in strict alphabetical order.

The field appears in the table on the left.

4. In the table on the left, in the Name column, enter a display name for the field.
5. Click OK.

FrameMaker Features You Need to Know

If you have not used FrameMaker before, do the Quick Tour in the FrameMaker User Guide to familiarize yourself with the product.

Before writing Wise documentation, make sure you know how to do the following in FrameMaker. Where noted, refer to the Wise Doc Procedures to learn about Wise-specific tasks.

- ◆ Open and work with a book.
- ◆ Add and delete chapters in the book.
- ◆ Update the book TOC and index.
- ◆ Turn change bars off and on for the entire book and for a single chapter. Also see [Using Change Bars](#) on page 91.
- ◆ Apply paragraph and character tags using the catalogs and using keyboard shortcuts.
- ◆ Import formats from another file to a book or a single file.
- ◆ Use conditional text: show and hide it, apply and remove it. Also see [Using Conditional Text](#) on page 110 and [Condition Tags in Single-Sourced Documents](#) on page 128.
- ◆ Work with index markers. Also see [Frame Indexing Issues](#) on page 36 and [Index for Help, Not Just the Book](#) on page 37. (You can read the rest of that chapter later, when you're ready to index.)
- ◆ Work with text insets. Also see [Text Insets](#) on page 133.
- ◆ Insert a screen shot or graphic in an anchored frame. Also see [Creating Screen Shots and Placing Graphics](#) on page 100.
- ◆ Use the graphic tools to create callouts in a graphic.
- ◆ Work with tables.
- ◆ Insert cross references. Also see [Cross-References](#) on page 28.
- ◆ Create a stand-alone Table of Contents and Index for a single chapter.
- ◆ Create a list of references to imported graphics.

Index

Numerics

- 5x8 template 74
- 7x9 template 74
- 8.5x11 template 74

A

- A4 page size
 - about 83
 - creating template 84
- Abreve 8
- Acrobat Distiller
 - setting up 166
 - West Wind Litho options 166
- acronym, indexing 39
- active voice 10
- Allegra Printing 177
- alternative words 20
- anchored frame
 - sizing to image 102
- anchored frame settings 102
- apostrophe, curly 73
- Application Gateway map IDs 186
- archiving
 - documentation 124
 - text insets 136
- article, in index 39
- audience 9
- author's note 112
- AutoMap
 - See Webworks AutoMap

B

- B1 BodyInBull1 tag 94
- B1 Bull1 tag 94
- B1B Bull1 Bold tag 94
- B2 BodyInBull2 tag 94
- B2 Bull2 tag 94
- B2B Bull2 Bold tag 94
- B3 BodyInBull3 tag 94
- B3 Bull3 tag 94
- B3B Bull3 Bold tag 94
- beta
 - documentation 148
 - help 149
 - manual 148
 - release notes 148
- bias-free writing 24
- book

- 5x8 format 160
- 7x9 format 161
- 8.5x11 format 162
- archiving 124
- changing view 158
- converting to new template 79
- numbering 155
- paginating 159
- PDF setup 156
- properties 155
- saving as PostScript 172
- sending to printer 175
- updating 160
- BookInfo plug-in 70
- BT BodyTight tag 95
- build, installing 233
- button menu, referring to 18, 53
- button, referring to 18, 53

C

- Callout tag 90
- capitalization, in index 40
- capture, screen
 - See screen shot
- caution 96
- caveat 96
- change bars
 - clearing 91
 - in selected text 92
 - searching for 92
 - turning on and off 91
 - when to use 91
- chapter number incorrect 226
- chapter opening 12
- character tag 89
- Character Tools plug-in 69
- checkbox, referring to 16, 53
- color, conditional text 130
- comment
 - in book 112
- condition template 131
- conditional document
 - archiving 127
 - managing 127
 - planning 126
 - template 131
 - text inset 133
 - variable 132
- conditional text
 - color 130
 - creating tag 129

- for anchored frames 128
 - hiding 157
 - shortcuts 111
 - tagging 128
 - template 131
 - using 110
 - what to hide 157
 - context-sensitive help
 - mapping 183
 - problems 226
 - testing 221
 - controls
 - formatting 25
 - indexing 47
 - referring to 16
 - conventions, typographic 25
 - conversion documents 78
 - convert document to new template 79
 - copying from Word to Frame 71
 - copyright
 - guidelines 156
 - in help 181
 - cross-book reference
 - adding 29
 - in index 40
 - cross-reference
 - forms 28
 - in index 41
 - to another Wise book 29
 - to SDK 29
 - to third-party help 29
 - unresolved 157
 - CudSpan Doc Tools
 - Hunt Overrides 70
 - MarkerWorker 70
 - curly quotes 73
- ## D
- data entry, referring to 17
 - deliverables
 - list 144
 - spreadsheet 225
 - to commercial printer 171
 - dialog name, indexing 47
 - Distiller
 - See Acrobat Distiller
 - documentation
 - final testing 225
 - preparing for review 146
 - documentation sets, Wise 114
 - documentation tools 69

drop-down list, referring to 17, 55

E

Edition tag 96
 edition, caveat 96
 Edition2 tag 97
 examples
 cultural considerations 21
 introducing 21
 extension, indexing 41
 external help topic, linking to 107

F

F1 help
 Also see context-sensitive help testing 221
 feature, indexing 45
 field entry, referring to 17
 file extension, indexing 41
 files
 locations 121
 sharing 122
 working doc files 121
 finishing steps checklist 153
 first person 10
 font
 missing font error 227
 footer, in help 182
 forced line break 87, 99
 forced return 87, 99
 format
 5x8 considerations 77
 A4 83, 84
 converting to 5x8 160
 converting to 7x9 161
 converting to 8.5x11 162
 formatting text 86
 formatting, special 25
 frame
 sizing to image 102
 Frame styles
 character 89
 creating 86
 FrameMaker
 anchored frame settings 102
 considerations 71
 graphic, placing 101
 image, placing 101
 missing font error 227
 personal dictionary 72
 plug-ins 69
 preferences 72
 setting up 72
 using a Mac 71
 FrameMaker templates
 5x8 74
 5x8 considerations 77

7x9 74
 8.5x11 74
 A4, creating 84
 about 74
 active 74
 changing 76
 conversion guidelines 78
 converting 79
 format differences 76
 obsolete 75
 future tense 10

G

gender-neutral writing 24
 graphic
 directories, cleaning 164
 in help 103
 placing in Frame 101
 preparing for help 182
 sizing for help 105

H

H4 Heading4, using 14
 header
 in help 182
 heading
 guidelines 13
 H4 Heading4 14
 help considerations 182
 line break 87
 single word 14
 syntax 44
 using 87
 help
 Also see WebWorks Publisher beta 149
 building 195
 context-sensitive 183
 context-sensitivity problems 226
 copyright and trademarks popup 181
 fonts, testing 220
 generating 195
 graphics 103
 hyperlinks, testing 221
 images 103
 importing to Source Safe 215
 index, testing 223
 mapping 184
 problems, reporting 223
 project specifications 196
 sizing images 105
 stand-alone 196
 test plan 220
 TOC, testing 222
 troubleshooting 226
 help ID
 See map ID
 help navigation
 adding to book 29

format 30

HelpFooter marker 182
 HelpHeader marker 182
 HTML Help, in Setup Editor updating 140
 Hunt Overrides plug-in 70
 hyperlink
 Also see help navigation testing in help 221
 to external help 107
 to SDK help 107
 to URL 106
 to Visual Studio help 108
 hyphenation, using 71

I

ID, help
 See map ID
 image
 in help 103
 placing in Frame 101
 preparing for help 182
 shrink-wrapping frame 102
 sizing for help 105
 imperative form 10
 incident, Littlebuggy 237
 Indented2Tight tag 95
 index
 formatting 160
 testing in help 223
 updating 160
 indexing
 acronyms 39
 considerations for help 37
 controls 47
 cross-book references 40
 dialogs 47
 feature names 45
 file extensions 41
 groups of items 42
 in Frame 36
 Installation Expert pages 45
 marker placement 37
 menu commands 46
 product name 43
 standards 32
 summary 34
 views (in UI) 46
 inline help
 linking to online help 141
 linking to SDK help 142
 Setup Editor 140
 testing 222
 inset
 See text inset
 Installation Expert page, indexing 45
 introductions 13

J

job, AutoMap
 creating 203
 scheduling 211

K

Kinko's 177

L

line break, forced 87, 99

link in book
 See cross-reference

list

bulleted 94
 non-bulleted 95
 numbered 94

list box, referring to 17

Littlebuggy

assignment code 239
 creating incident 237
 creating query 239
 customizing display 240
 e-mailing from 236
 sample queries 239
 using 236

M

map ID

Application Gateway 186
 file location 183
 finding in product 183
 finding with RoboHelp 184
 in inline help 186
 in SetupEditor Tables tab 185
 mapping 184
 Wise custom script actions 186

markers, editing 70

MarkerWorker plug-in 70

menu command, indexing 46

menu, referring to 19

missing font (Frame) 227

multi-lingual production

requirements 113

multi-tab dialog, referring to 19

N

navigation, help
 see help navigation

not for help, condition tag 110

note

author's note 112
 in documentation 96

noun, in index 42

numbered list 94

Numbered tag 94

numbering

resetting 155

troubleshooting in Frame 226

O

overrides, finding 70

P

Package Distribution options,
 enabling 231

page (Installation Expert),
 indexing 45

page count, getting for book 70

page range, in index 42

page size, A4

about 83
 creating template 84

paginating book 159

PaintShop Pro, for screen shots 101

Paragraph Tools plug-in 69

passive voice 10

past tense 10

pasting from Word to Frame 71

PDF

creating 167
 for quick printing 177
 settings 169
 setup in book 156
 troubleshooting 228

person 10

phone number 51

plug-ins for Frame 69

PostScript

creating for book 172
 driver, installing 172
 driver, setting up 172
 testing 174

preferences

AutoMap 202
 FrameMaker 72
 WebWorks Publisher 188

preposition, in index 42

present tense 10

printing book

for proofing 174
 sending to printer 175

procedure

introducing 14
 writing 22

product

installing 233

product edition, caveat 96

product name variable 132

product name, indexing 43

production requirements, multi-
 lingual 113

pronouns, neutral 24

proof, reviewing 175

properties

book 155
 Postscript driver 172

Q

query, Littlebuggy

creating 239
 customizing display 240
 samples 239

quick printing

creating .PDF 177
 specifications 178

R

radio button, referring to 16

readme

See release notes

redirection, in index 43

release notes

beta 148
 creating 152
 guidelines 151
 links to previous notes 151

review documentation

guidelines 146
 preparing 146

right-click menu, referring to 18

S

S1 Script1 tag 89

S2 Script2 tag 89

S3 Script3 tag 89

S4 Script4 tag 89

sample screen

See screen shot

screen colors, for screen shots 100

screen shot

colors 100
 guidelines 100
 placing in Frame 101
 using PaintShop Pro 101
 when to use 27

Script tag 89

SDK help

cross-reference 29
 linking from inline help 142
 linking to 107

second person 10

sentence construction 10

Setup Editor help, updating 140

sharing Frame files 122

Silicon Prairie Software

Character Tools 69
 Paragraph Tools 69

single-sourced book

archiving 127
 managing 127
 planning 126

- template 131
 - text inset 133
 - variable 132
- smart quotes 73
- smart spaces 73
- space between sentences 73
- special character, in index 44
- stand-alone help 196
- standards for writing 8
- style tags (Frame)
 - character 89
 - creating 86
 - deleting 77
- syntax for topic headings 44
- Systemc Toolbox plug-in 70

T

- table
 - anchoring 98
- table of contents
 - editing in book 160
 - testing in help 222
 - updating 160
- tabs, referring to 19
- template
 - help 188
 - help, creating 189
 - WebWorks 188
 - WebWorks, creating 189
- templates
 - 5x8 74
 - 5x8 considerations 77
 - 7x9 74
 - 8.5x11 74
 - A4, creating 84
 - active 74
 - changing 76
 - conversion guidelines 78
 - converting 79
 - format differences 76
 - FrameMaker 74
 - obsolete 75
- tense 10
- terminology standards 51
- testing
 - documentation 225
 - help 220
- text inset
 - about 133
 - archiving 136

- creating 134
 - guidelines 133
 - importing 134
 - problems 135
- text, formatting 86
- time, reporting 235
- timesheet 235
- tip, in documentation 96
- TOC
 - See table of contents
- Toolbox plug-in 70
- tools for documentation 69
- trademarks, in help 181
- troubleshooting
 - AutoMap 212
 - context-sensitive help 226
 - extra pages 227
 - missing fonts 227
 - numbering 226
 - PDF 228
- tutorial files in product 226
- typographer's quotes 73
- typographic conventions 25

U

- unnecessary words 20
- unresolved cross-references 157
- updating book 160
- URL
 - creating link in book 106
 - format in book 28
- user profile 9
- utilities for Frame 69

V

- variable, product name 132
- verb form, in index 48
- verb, in index 48
- version, guidelines 156
- views (in UI)
 - indexing 46
 - referring to 19
- Visual Studio .NET 137
- Visual Studio help, linking to 108
- Visual Studio Integrated Editor 137
- voice
 - active 10
 - passive 10

W

- warning 96
- WB Inline Help template 194
- WebWorks AutoMap
 - file locations 201
 - importing to SourceSafe 215
 - installing 200
 - jobs 203
 - preferences 202
 - scheduling 211
 - troubleshooting 212
- WebWorks Publisher
 - Also see help
 - generating help 195
 - preferences 188
 - project specifications 196
 - template, about 188
 - template, creating 189
- West Wind Litho
 - Acrobat job options 166
 - creating .PDF 177
 - delivering files 179
- WideBody tag 90
- WideGraphic tag 90
- Wise documentation sets 114
- Wise for Visual Studio .NET
 - documentation 117
- Wise for Windows Installer
 - documentation set 116
 - uninstalling 233
- Wise Help template 191
- Wise Installation System
 - documentation set 118
- Wise InstallTailor documentation 119
- Wise Package Studio
 - documentation set 115
 - uninstalling 234
- Wise SDK template 191
- word count, getting for book 70
- Word, copying and pasting from 71
- Workbench tools, describing 22
- working files 121
- WPS Help template 191
- WPS Web Help template 192
- writing a procedure 22

X

- XH1 Heading1 87