March 14, 2000

Update: Single-Sourcing from FrameMaker 5.5.6 to ForeHelp Premier 2000

Deborah Snavely

This article centers on a single-source documentation process that involves authoring structured, chunked documentation in Adobe FrameMaker 5.5.6 and converting it “just in time” to a WinHelp online help system using ForeHelp Premier 2000.* This process depends both on Adobe FrameMaker templates and ForeHelp’s new MIF import capability.

This article presumes you understand how to use FrameMaker templates and book files, and have some knowledge of online help projects or help-authoring tools.

Single-Sourcing—Write Once, Read Many

Traditional software development cycles are shortening, especially in young companies. Web software and content, in particular, has a half-life of days or weeks, not years. Meanwhile, the historical headcount squeeze has been replaced by a talent shortage. All these trends increase pressure on Publications departments—especially in small firms or departments—to write documentation content only once per product and release, and deliver that content in as many formats as required:† print, PDF, online help, and Web site.

* The parent of this process was developed by Deborah Snavely and Ed Rush at SmallFirm Technology in 1996, using the the then-current FrameMaker 5.1.1 and ForeHelp 2.96. An article describing it appeared appeared online in the December 1999 issue of InFrame magazine (http://www.inframe-mag.com/) Special acknowledgments are due Ed Rush for his assistance with parts of the source material for both articles.

† Such a write-once, read-many goal has been possible for years, of course—with a big up-front investment in software, process improvement, consultants, and training—using SGML and related tools.

Conversions

Experience has taught us all that converting content from one application to another carries an overhead cost. It’s never as simple as our software tools promise. When you must perform such conversions routinely, the limitations of both products—the conversion source and the conversion destination—usually apply to the resulting process.

Single-sourcing, by definition, demands significant conversion capabilities. (The work that determines content success is in the pre-writing: researching, analyzing, planning, designing, prototyping, and testing. The production work—the conversion—needs to be as simple and painless as possible, so that Pubs can concentrate on continuing the content improvement for the deliverable outputs.) Fortunately, FrameMaker and ForeHelp provide conversion support on both sides of the process.

Output

Setting up a conversion process requires prototyping to learn the best settings for your routine conversions. They may also require some compromises before or after conversion: of format, of writing style, of graphics handling. In the case described here, I used my knowledge of an older single-source procedure (FrameMaker to RTF to ForeHelp) to convert a large document into an online help system using ForeHelp’s newly released MIF import feature. I expected several potential benefits from the new feature:

- clean formatting transfer of tagged paragraphs and characters
- automated import of graphics in anchored frames
- conservation of graphic file names for graphics imported by reference
- automated conversion of FrameMaker-created graphics

Origin of the Process

As a two-person Pubs department at SmallFirm Technology in 1996, Ed Rush and I collaborated on a method for creating SmallFirm Technology software documentation in FrameMaker and converting it to ForeHelp just in time to build the beta online help system for a product release. Of course, this method demanded that we structure and write the document from the outset in ways that supported the planned conversion:

- develop structure first
- use standard mini-structures (procedures, reference content, and introductions) to create parallelism among sections (topics) and between writers
- create bulleted cross-reference lists to introduce subordinate content
- divide product features and commands into task-oriented sections (which become individual help topics)
- write “chunked” content to keep topics appropriately brief
  - no heading without a paragraph
  - no paragraph without a heading
  - only one topic per heading (one paragraph or “chunk,” which might include a note, tip, bullet list, procedure, figure, or table)

Testing Under Fire

With variations, this method saw me through a number of single-source efforts. With the recent release of ForeHelp Premier 2000, however, I needed to test its ability and reliability of the new import feature: MIF import. At the same time, I needed to do a fast update of an online help system that derived from a FrameMaker document which had been massively overhauled since the content was moved into an existing online help system.

Could ForeHelp 2000 do the job? Based on ForeHelp's track record of solid, prompt technical support, I decided to risk a live test. I expected to need less preparation in FrameMaker, but I had no conversion-prep templates prepared. I worked from memory, and referred to my previous article to keep me on track.

The following outline describes both ends of the FrameMaker-to-ForeHelp conversion process in two major sections:

- Working in FrameMaker
- Working in ForeHelp

Working in FrameMaker

Before the conversion, I used FrameMaker's capabilities to prepare the files:

1. Finalize the FrameMaker book as if for production, including all generated files (table of contents, lists of figures and tables, index).
2. Archive a copy of the project directory for backup before you start the conversion (if you work on a server, make a local copy for the conversion).
3. Change cross-reference formats to remove chapter numbers.
4. Check graphics for Frame-added callouts. For those graphics only, enlarge the layout, hide text symbols and borders, zoom in so that figure fills the screen, take a screen shot, and crop and save the online version as a GIF file.
5. Export each body file into MIF format.

Working in ForeHelp

ForeHelp enable a lot more automation in the conversion:

1. Create a new help project from the template project in ForeHelp. (If you haven't got a template project, import one MIF chapter as a test and set up the style names, styles, and default project settings as a template project.)
2. Select a body MIF file to import, and, in the import file dialog box, and adjust the settings. I used the following settings:
   - In the General tab: create keywords from index entries, discard unused styles, include anchored frames in anchoring topic, use ForeHelp styles
   - In the Style Options tab: check each FrameMaker tag to import and map it to the ForeHelp output style you want; set which FrameMaker tags should start new topics; and, if they do, whether they should become topic banners, be added to the ForeHelp contents file, or be added to topic keywords

3. Import each MIF file into the help project. (Refer to printout and adjust import settings to start new topics appropriately for the chapter contents.)

   Note: Save early and often.

4. Link the entries in the ForeHelp contents topic to the individual topics.

How Did It Go?

For starters, I noticed all the steps I didn't need! Beyond that, what I mostly found was a seamless experience. My source FrameMaker files were not of my writing: they were written by someone on leave and edited by another writer—I only converted them. So I couldn't rely on intimate knowledge of the contents, I had to trust that the other writers used our FrameMaker tags and templates as expected. (Yes, they did! ForeHelp lets you delete format overrides during import, if you wish.)

The Good News

Lots of good news. It was so smooth I couldn't believe it was a *.00 release! (Kudos to the beta testers.) Tags behaved, tables stayed put, text flowed into banners, graphics were pretty much where I wanted them...amazing.

- Graphics
  Graphics came over beautifully: different formats, etc. Many of our screen shots are GIFs, though older ones are BMPs; both came in without a grumble. Graphics that had been copied into Frame (instead of imported by reference) were automagically named things like MIF123.bmp or MIF789.gif and placed in the help project's images directory, whatever you've specified. (You need to specify source file, graphic file, and build directories in the template project before importing the MIF files.) Graphics imported by reference retained their own names, making for easy replacement in case of last-minute changes to a product screen. With graphics imported automatically, I omitted figure as well as table titles from the paragraphs selected to import into ForeHelp instead of deleting them from the project afterwards.
- Tables
  No problem, with one exception, noted below. It may be a bug, but it may have been corruption in the FrameMaker file. I haven't researched the problem further, and ForeFront, Inc. has my sample in hand for experiments.

The Bad News

Actually, there wasn't much. Certainly nothing that prevented my conversion from staying brisk.

- FrameMaker-generated graphics
  I found that ForeHelp imported all the callouts to a Frame-commented graphic as untitled popup topics and ignored the callout lines and only placed the BMP file the callouts documented. I didn't have time to test a suite of variations and see whether any of the Frame-created graphics imported correctly; I fell back on my 1996 workaround to the same issue: make the FrameMaker-assembled graphic as nice as possible, enlarge onscreen with no borders or symbols, and take a screen shot of *that* and crop it to size, then replace the incomplete image in the ForeHelp project. I'll be

looking forward to testing the parameters of this behavior later. No, it's not perfect; but I had to do a lot more of that in my previous single-source process.

- Multiply layered conditions
  I did uncover an apparent bug: one particular topic was incomplete, breaking off in the middle of a specific table (the MIF file imported correctly; only that partial topic was incomplete). When I examined the source Frame file, I found that the break point occurred in the middle of a conditional table row which *also* contained conditional text. With time my enemy, I fell back on a FrameMaker workaround: I copied the section that contained the table into a separate Frame file, turned off conditions, and deleted everything that wasn't supposed to show, then exported the MIF again. However, the problem may have been a corrupted FrameMaker table, because I had to convert the table to paragraphs and back into a table before the MIF export worked correctly. There may even have been some other hidden FrameMaker artifact that I didn't notice. I'll let the ForeHelp folks figure it out.

Conclusion

I'm still learning my way around ForeHelp's new features; everything I've learned to date supports my single-source goals. At the same time, Adobe has just (literally, yesterday!) announced the release of FrameMaker 6.0 for April 2000 on all platforms. As the quality gurus' mantra has it, improvement is a continuiing process and not something one does and then stops. Here's hoping that the new FrameMaker release shows as much sound attention to usability and the needs of the user community as this release of ForeHelp does.