Outside of a dog, a book is man's best friend.
Inside of a dog it's too dark to read.
—Groucho Marx

PREFACE

FOSI is the native formatting language used by Arbortext Editor for the Edit window display and print/PDF output. Styler™ and XSL-FO stylesheets are ultimately mapped to FOSI but do not support all the formatting capabilities available with native FOSI.

FOSI is tightly integrated with the Arbortext product line, which enables development of powerful Arbortext Editor applications that run at top speed. For example, line numbering requires FOSI, as does Arbortext Editor's change processing application.

This book is entitled *Practical FOSI* because it focuses on a practical approach to utilizing the various mechanisms FOSI provides for specifying formatting. Theory is left to the FOSI standard, (*MIL-PRF 28001C Appendix B*). The goal here is to facilitate development and maintenance of production FOSI stylesheets.

NOTE: The "practical FOSI" methodology is tried and true. It has been successfully used by FOSI stylesheet developers for almost twenty years.

TRIVIA

Search the Arbortext distributed files for `*.fos` to see that FOSI is used to format many aspects of the product line.

Audience

Practical FOSI is designed for those who want to:

- develop a FOSI stylesheet
- maintain a FOSI stylesheet
- make source edits in Styler™
- understand what FOSI can contribute to an Arbortext Editor application

Prerequisites

A working knowledge of structured markup, Arbortext Editor, and DTDs is a prerequisite for learning FOSI formatting. Programming experience, however, is not required. The most useful background is desktop publishing, typesetting, or other document preparation experience.

What's in this book

Practical FOSI covers the following:

- an overview of FOSI formatting
- information on the “Practical FOSI” approach to FOSI stylesheet development
- FOSI reference information
- advanced coding techniques
- how-to instructions
- a methodology for developing FOSI stylesheets
- information on FOSI troubleshooting
- Arbortext Command Language related to FOSI formatting
- tips and tricks

The appendices contain the following:

- supplementary information about formatting for paper
- Arbortext's version of the OutSpec DTD with annotations
- Other resources available to FOSI developers

What's not in this book

The FOSI standard, officially titled *MIL-PRF 28001C Appendix B*, is not reproduced in *Practical FOSI*. The FOSI standard was written for software engineers developing support for FOSI, not to provide user documentation.

Practical FOSI does not contain tutorials and does not cover Arbortext Editor's two powerful FOSI interfaces, the style panels and the tagged editor. For that, you need the author's FOSI tutorials, which are designed so you can learn FOSI as quickly and easily as possible. The tutorials teach important concepts while providing lots of useful how-to coding examples and teaching the use of the two FOSI development interfaces. After completing the tutorials, you will understand how FOSI works, have some familiarity with everything it can do in Arbortext Editor, and know how to make the most of the FOSI development interfaces so you can hit the ground running. In addition, with its tips and tricks, FOSIs, and test documents, the tutorials are an ongoing resource. Please visit the author's web site at `www.FOSIexpert.com\tutorials.html` for more information.

NOTE: *Practical FOSI* does not document Arbortext's Change Pages Application (CPA) or any FOSI components used only by CPA. Additionally, this book does not document Publishing Engine.

Conventions used in this book

The following conventions are used in *Practical FOSI*:

- The phrase SGML/XML is used because FOSI formats both.
- “Formatter” and “formatting engine” are used because it is not practical to distinguish Arbortext Editor, Print Composer, and Publishing Engine.
- Since some characteristics apply only to composed (print/PDF) output and have no effect on the Edit window display,  is used to indicate at least one of the category's characteristics is supported for Edit window display, while  indicates at least one of its characteristics is supported for print/PDF output.
- In OutSpec DTD fragments, a double underscore indicates the formatting is supported for both Edit window display and print/PDF output. A single underscore means the formatting is supported only for print/PDF.

- Document elements are shown with angle brackets. For example: <note>. Attributes shown separately are in italic. For example: *role*.
- FOSI categories and characteristics are shown in a monospaced font. For example: `algroup` and `refpoint`.
- File/path names and values entered by the user also use a monospaced font. For example: `10pi+4.5pt`.
- Footnotes in this book follow a sequence of symbols, starting with a dagger (†). The asterisk (*) is not used because it has meaning in DTDs and the FOSI language.

About examples

Just as a good picture is worth a thousand words, so is a good example. *Practical FOSI* contains many examples to demonstrate FOSI's power as well as provide code you can use and inspiration for code you can develop yourself.

NOTE: The examples were tested in version 5.3 M030 of Arbortext Editor with Print Composer.

Examples are generally structured as follows:

1. A graphic of the formatted output or of the Edit window display is shown first.
2. When necessary, a DTD fragment is provided next. Otherwise, assume the markup and structure shown in the example are valid.
3. The relevant XML or SGML fragment is next, unless the graphic shows the Edit window or the input is obvious.
4. Last but not least is the FOSI fragment. FOSI code that is the essence of the example is highlighted so it is easy to spot.

NOTE: FOSI fragments in examples often reference charsubsets whose definitions are not included in the FOSI fragment. These charsubsets are defined in **Figure 7 Useful `charsubsets`** on page 30.

NOTE: Smaller than normal page dimensions are often used in *Practical FOSI* to illustrate page-related formatting in a reasonable amount of space. Similarly, a larger than usual font size may be employed in sample output so the object of discussion is easy to see without magnification.

NOTE: To conserve space, end tags for FOSI code examples may be omitted when their existence can be inferred. For instance, in the following FOSI code,

the end tags </charlist> and </e-i-c> are implied by ellipsis dots (...), which suggest that other categories may be coded in the e-i-c. In addition, FOSI code fragments may assume the existence of supporting code, such as the charsubsets referenced in this example.

```
<e-i-c gi="emphasis">
<charlist inherit="1" charsubsetref="inline bold">
...
```

NOTE: The FOSI code examples in this book assume the document in question is valid and complete. When a document is out of context and when there are missing and/or duplicate IDs, FOSI formatting may not be correct.

Acknowledgments

...

Colophon

This book was authored using Arbortext Editor with the author's EZ interface and a DocBook-based custom DTD. It was formatted for print using Print Composer and a custom FOSI.

Century Schoolbook and Century Gothic fonts were chosen for text and titles, respectively, because they provide serif and sans serif fonts that go well together.

Cover photo: "Summer Heat," James Stewart, ©2011.