XML HACKS™

100 Industrial-Strength Tips & Tools

O'REILLY®

Michael Fitzgerald

HACK #65 Unravel the OpenOffice File Format

OpenOffice provides a suite of applications whose native file format consists of a set of XML files, compressed into a ZIP archive. This hack explores the basics of the OpenOffice file format.

OpenOffice (*http://www.openoffice.org*) is a suite of free, multiplatform, open source applications for the desktop, sponsored by Sun Microsystems (*http://wwws.sun.com/software/star/openoffice/*). The suite includes text-editor, spreadsheet, drawing, and presentation applications, each of which uses an XML-based file format. Table 4-2 lists the OpenOffice applications and their file extensions.

Each file is saved as a collection of XML documents and stored in a ZIP archive. (You can also save documents in other formats, such as text, Rich Text Format, or HTML. You can also export a document as PDF.) The specification of the OpenOffice XML file format is being maintained by an OASIS technical committee (*http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=office*).

Table 4-2. OpenOffice applications and file extensions

OpenOffice application	File extension
Calc spreadsheet application	`*.sxc`
Calc templates	`*.stc`
Draw graphics application	`*.sxd`
Draw templates	`*.std`
Impress presentation application	`*.sxi`
Impress templates	`*.sti`
Math application	`*.sxm`
Master files	`*.sxg`
Writer text editor application	`*.svw`
Writer templates	`*.stw`

In the *OpenOffice* subdirectory of the book's file archive is a small file, *foaf.sxw*, a snippet taken from the FOAF hack **[Hack #64]**. It is shown in OpenOffice's Writer application in Figure 4-5. You can use any ZIP tool to examine or extract the XML files from this ZIP file. I'll use the *unzip* command-line tool that comes with Unix distributions such as Cygwin (*http://www.cygwin.org*).

While in the *OpenOffice* subdirectory, enter this command at a shell prompt:

```
unzip -l foaf.sxw
```

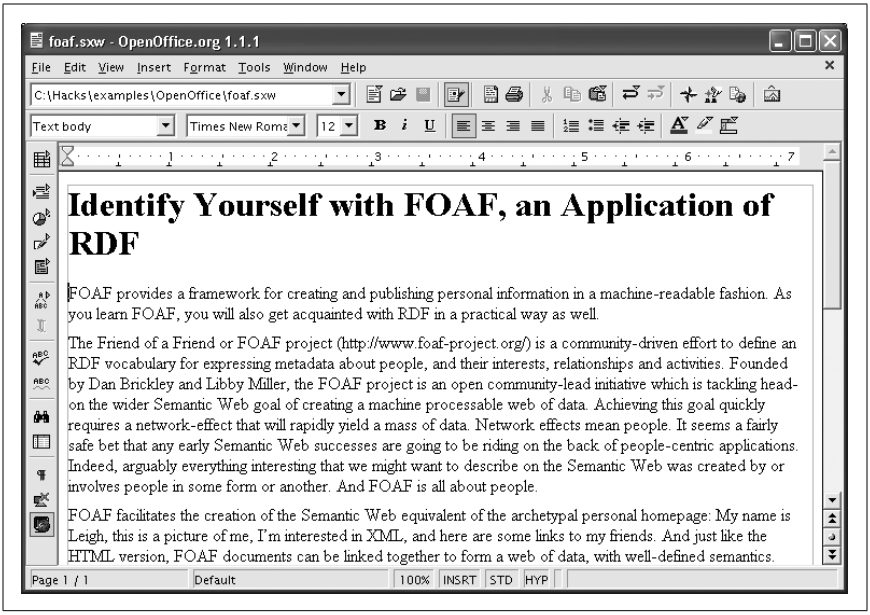

Figure 4-5. foaf.sxw in OpenOffice's Writer application

The `-l` option allows you to inspect the contents of the compressed file without extracting the files from it. This command produces:

```
Archive:  foaf.sxw
  Length     Date   Time    Name
 --------    ----   ----    ----
       30  04-04-04 04:51   mimetype
     4178  04-04-04 04:51   content.xml
     8062  04-04-04 04:51   styles.xml
     1174  04-04-04 04:51   meta.xml
     9180  04-04-04 04:51   settings.xml
      752  04-04-04 04:51   META-INF/manifest.xml
 --------                   -------
    23376                   6 files
```

Extract these files into the *OpenOffice* subdirectory with:

```
unzip foaf.sxw
```

You'll see this:

```
Archive:  foaf.sxw
 extracting: mimetype
  inflating: content.xml
  inflating: styles.xml
 extracting: meta.xml
  inflating: settings.xml
  inflating: META-INF/manifest.xml
```

Briefly, here's what each of these files contains:

mimetype
: Contains the file's media type; e.g., `application/vnd.sun.xml.writer`.

content.xml
: Holds the text content of the file.

meta.xml
: Holds any meta information for the document. You can edit the meta information associated with this document by selecting File → Properties.

settings.xml
: Contains information about the settings of the document.

styles.xml
: Stores the styles applied to the document. You can apply styles to the document by selecting Format → Stylist (or by pressing F11).

META-INF/manifest.xml
: Contains a list of XML and other files that make up the default OpenOffice representation of the document.

When you do a File → Save As, you can click the "Save with password" checkbox. If you do this, all the XML files except *meta.xml* are saved as encrypted files.

For illustration, we'll look at one of the files stored in the OpenOffice saved-file archive. Example 4-12 shows the XML markup that's inside *content.xml*. This document is nicely indented because in the Tools → Options Load/Save dialog box under General settings, I've unchecked the Size optimization for XML format (no pretty printing) checkbox. It's checked by default, meaning that normally the XML files are saved without indentation.

Example 4-12. content.xml from foaf.sxw

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE office:document-content PUBLIC
"-//OpenOffice.org//DTD OfficeDocument 1.0//EN" "office.dtd">
<office:document-content
 xmlns:office="http://openoffice.org/2000/office"
 xmlns:style="http://openoffice.org/2000/style"
 xmlns:text="http://openoffice.org/2000/text"
 xmlns:table="http://openoffice.org/2000/table"
 xmlns:draw="http://openoffice.org/2000/drawing"
 xmlns:fo="http://www.w3.org/1999/XSL/Format"
 xmlns:xlink="http://www.w3.org/1999/xlink"
 xmlns:number="http://openoffice.org/2000/datastyle"
 xmlns:svg="http://www.w3.org/2000/svg"
```

Example 4-12. content.xml from foaf.sxw (continued)

```
 xmlns:chart="http://openoffice.org/2000/chart"
 xmlns:dr3d="http://openoffice.org/2000/dr3d"
 xmlns:math="http://www.w3.org/1998/Math/MathML"
 xmlns:form="http://openoffice.org/2000/form"
 xmlns:script="http://openoffice.org/2000/script"
 office:class="text" office:version="1.0">
 <office:script/>
 <office:font-decls>
  <style:font-decl style:name="Tahoma1" fo:font-family="Tahoma"/>
  <style:font-decl style:name="Lucida Sans Unicode"
   fo:font-family="&apos;Lucida Sans Unicode&apos;"
       style:font-pitch="variable"/>
  <style:font-decl style:name="MS Mincho"
       fo:font-family="&apos;MS Mincho&apos;"
   style:font-pitch="variable"/>
  <style:font-decl style:name="Tahoma" fo:font-family="Tahoma"
   style:font-pitch="variable"/>
  <style:font-decl style:name="Times New Roman"
   fo:font-family="&apos;Times New Roman&apos;"
       style:font-family-generic="roman"
   style:font-pitch="variable"/>
  <style:font-decl style:name="Arial" fo:font-family="Arial"
   style:font-family-generic="swiss" style:font-pitch="variable"/>
 </office:font-decls>
 <office:automatic-styles>
  <style:style style:name="P1" style:family="paragraph"
   style:parent-style-name="Text body">
   <style:properties fo:text-align="center"
        style:justify-single-word="false"/>
  </style:style>
  <style:style style:name="fr1" style:family="graphics"
   style:parent-style-name="Graphics">
   <style:properties style:vertical-pos="top"
        style:vertical-rel="paragraph"
    style:horizontal-pos="center" style:horizontal-rel="paragraph"
    style:mirror="none" fo:clip="rect(0inch 0inch 0inch 0inch)"
        draw:luminance="0%"
    draw:contrast="0%" draw:red="0%" draw:green="0%" draw:blue="0%"
        draw:gamma="1"
    draw:color-inversion="false" draw:transparency="0%"
    draw:color-mode="standard"/>
  </style:style>
 </office:automatic-styles>
 <office:body>
  <text:sequence-decls>
   <text:sequence-decl text:display-outline-level="0"
          text:name="Illustration"/>
   <text:sequence-decl text:display-outline-level="0"
          text:name="Table"/>
   <text:sequence-decl text:display-outline-level="0"
          text:name="Text"/>
```

Example 4-12. content.xml from foaf.sxw (continued)

```
    <text:sequence-decl text:display-outline-level="0"
           text:name="Drawing"/>
   </text:sequence-decls>
  <text:h text:style-name="Heading 1" text:level="1">Identify Yourself with FOAF,
  an Application of RDF</text:h><text:p text:style-name="Text body">
  FOAF provides a framework for creating and  publishing personal information
  in a machine-readable fashion. As you learn FOAF,  you will also
  get acquainted with RDF in a practical way as well.</text:p>
  <text:p text:style-name="Text body">The Friend of a Friend or FOAF project
 (http://www.foaf-project.org/) is a community-driven effort to define an RDF
  vocabulary for expressing metadata about people, and their interests,
  relationships and activities. Founded by Dan Brickley and Libby Miller, the FOAF
  project is an open community-lead initiative which is tackling head-on the wider
  Semantic Web goal of creating a machine processable web of data. Achieving this
  goal quickly requires a network-effect that will rapidly yield a mass of data.
  Network effects mean people. It seems a fairly safe bet that any early Semantic
  Web successes are going to be riding on the back of people-centric applications.
  Indeed, arguably everything interesting that we might want to describe on the
  Semantic Web was created by or involves people in some form or another. And FOAF
  is all about people.</text:p><text:p text:style-name="Text body">
   FOAF facilitates the creation of the Semantic Web equivalent of the
  archetypal personal homepage: My name is Leigh, this is a picture of me,
  I'm interested in XML, and here are some links to my friends. And
  just like the HTML version, FOAF documents can be linked together to form a web
  of data, with well-defined semantics.</text:p><text:p text:style-name=
  "Text body"> Being a W3C Resource Description Framework or RDF application
  (http://www.w3.org/RDF/) means that FOAF can claim the usual benefits of being
   easily harvested and aggregated. And like all RDF vocabularies, it can be
  easily combined with other vocabularies, allowing the capture of a very rich set
  of metadata. This hack introduces the basic terms of the FOAF vocabulary,
  illustrating them with a number of examples. The hack concludes with a brief
  review of the more interesting FOAF applications and considers some other uses
  for the data. The FOAF graphic is shown in Figure A-1.</text:p>
  <text:p text:style-name="P1">Figure A-1: FOAFlets</text:p>
  <text:p text:style-name="Text body"/>
  <text:p text:style-name="Text body">
  <draw:image draw:style-name="fr1"
  draw:name="Graphic1" text:anchor-type="paragraph" svg:width="4.2201inch"
  svg:height="2.4299inch" draw:z-index="0"
  xlink:href="#Pictures/10000000000001A6000000F34FFA992C.jpg"
  xlink:type="simple"xlink:show="embed" xlink:actuate="onLoad"/></text:p>
  </office:body>
</office:document-content>
```

The XML documents in OpenOffice use DTDs **[Hack #68]** that come with the installed package, though XML Schema and RELAX NG schemas will be available in future versions. For example, on Windows, these files are installed by default in *C:\Program Files\OpenOffice.org1.1.1\share\dtd\officedocument\1_0*. This document uses *office.dtd* (line 3). (These DTDs are not in the book's file archive.) On line 4, the `office:document-content` ele-

ment is the document element with the namespace `http://openoffice.org/2000/office`. Many other namespaces are declared, along with some familiar ones, such as for SVG **[Hack #66]** and XSL-FO **[Hack #48]**.

Various font declarations are stored in `style:font-decl` elements on lines 21 through 37. Attributes with the `fo:` prefix properties from XSL-FO. Lines 38 through 56 list styles that are used in the document. Lines 58 to 67 contain markup used for numeric sequencing in the document. A heading appears on line 68, followed by body text in lines 69 through 97. Lines 98 through 106 show how OpenOffice defines a reference to a graphic, including attributes from the SVG and XLink namespaces such as `svg:width` and `xlink:href`. The embedded graphic is stored in the *Pictures* subdirectory of *foaf.sxw* as the file *10000000000001A6000000F34FFA992C.jpg* (line 104).

See Also

- For details on the OpenOffice file format, see the OASIS OpenOffice specification: *http://www.oasis-open.org/committees/download.php/6037/office-spec-1.0-cd-1.pdf*
- For documentation and examples of working with OpenOffice XML, see J. David Eisenberg's *OpenOffice.org XML Essentials* (*http://books.evc-cit.info/*)