

Select Your Own Adventure Interpreter

The Exciting README Document

Luke Gane

What It Is

A SWF file that can interpret a simple XML file containing the contents of a basic "choose select your own adventure"-style book and display it in an interactive format that's easy for people to read through.

How It Works

There are three components to the SYOA Interpreter: the SWF file containing the interpreter itself, the XML file containing the contents of the book, and some HTML code to embed the interpreter in a web page. The XML file must be in the same directory as the SWF file. The file name of the XML file you wish the SWF file to load and interpret is specified in the HTML code using flashVars:

```
...
<param name="flashVars" value="xmlFilename=syoa_book%2Exml" />
...
<embed src="syoa_interpreter.swf"
flashVars="xmlFilename=syoa_book%2Exml" ...
...
```

(Note that %2E is the percent-escaped code for a period.)

For more detail, see the included example HTML file *syoa_interpreter.html* (note that it's pretty much the one generated by Flash 8 with the flashVars stuff added, so the coding style is not recommended).

For more information about using flashVars, see:
<http://www.permadi.com/tutorial/flashVars/index.html>

For a table of ASCII character codes (if you like complex filenames), see:
http://www.w3schools.com/tags/ref_urlencode.asp

Structure of the XML File

You can probably get a pretty good idea of how the XML file should be structured from the included example file *syoa_book.xml*, but there are a few things that might require a bit of explanation.

- **Order of nodes:** within a page node, the description node must come first, followed by a maximum of three option nodes. Any options after the first three will either be ignored or crash the code when it tries to interpret the XML. The maximum number of options was set at three due to space constraints; hopefully it's sufficient. Also note that, if your description text is extremely verbose, options

may be pushed off the bottom of the screen. If this happens, read "The Elements of Style" and try again.

- **The title_size attribute:** This allows you to specify the font size (in pts) of the title text. This is useful if you have a long title that's going beyond the grey bars on the title screen.
- **CDATA:** the text for the description and option nodes is enclosed in CDATA tags so Flash doesn't get confused.
- **Page numbers and link pages:** the number attribute for a page specifies the page's page number (hopefully this is obvious); note that the interpreter always starts at page 1. (You can have a page 0 if you really want to, though.) The link_page attribute for option nodes specifies the page the interpreter should turn to when the reader selects that option.

Overall, the basic structure of the XML file is:

```
<?xml version="1.0" ?>
<adventure title="TITLE GOES HERE" title_size="NUMBER GOES HERE"
author="YOUR NAME OR ENIGMATIC ALIAS GOES HERE">
  <page number="NUMBER GOES HERE">
    <description>
      <![CDATA[PAGE DESCRIPTION GOES HERE]]>
    </description>
    <option link_page="NUMBER GOES HERE">
      <![CDATA[OPTION TEXT GOES HERE]]>
    </option>
    <option link_page="NUMBER GOES HERE">
      <![CDATA[OPTION TEXT GOES HERE]]>
    </option>
  </page>
</adventure>
```

Miscellaneous Comments (2015)

I made this interpreter several years ago using Macromedia Flash 8 and Actionscript 2. Rather remarkably, Flash 8 (released in 2005) still runs on 64-bit versions of Windows 7, although I think it's fair to say it would be a rather poor choice for any form of software development at this point (Actionscript 3 and FlashDevelop being decidedly more sensible solutions for the modern Flash programmer).