Apache™ FOP Design: Extensions

Version 1298724

by Keiron Liddle

Table of contents

1 Introduction

Apache™ FOP provides an extension mechanism to add extra functionality. There are a number of different types of extensions that apply to different steps when converting FO into the rendered output.

2 Extensions

SVG Graphic - This applies to svg and any other xml document that can be converted into svg in the output. All that is required is the element mapping for the xml and a converter that changes the document into svg. This conversion is done in the FO Tree. The conversion is done by the top level element of the namespace or in the case of an external image a Converter.

XML Document - Instead of converting the document into svg it can be passed directly to the renderer. The renderer will need to have a handler for the xml document. This handler can add information directly to the output document.

Output Document - This is used to add document level information to the output result. Such an extension will set information that is passed to the output document. The area tree handles these extensions and passs along the information to the renderer. The extension may contain resolveable objects. The extension can be passed to the renderer once resolve either immediately, after the next page or at the end of the document. This is so that the extension can be handled according to other associated data.

FO Area - This is where an extension creates an normal or extended area in the Area Tree. This is useful when the normal FO objects cannot create the area in the way that is needed.

Resolveable - In some cases it may require information to be resolved for information such as page numbers. This can apply to the XML Document, FO Area or output document extensions.

- Add a string ['(Continued)'] to a table header if the table spans multiple pages. These tables are part of the content and can start anywhere in the page.
- Separate page number display for a subsection. ie. - master document is page 4 of 7, but subsection is page 2 of 3.

3 Examples

Plan - The plan extension is a simple SVG graphic extension. Given a plan document either inside an InstreamForeignObject or as an external graphic, it converts the plan document into an svg graphic. The svg graphic is then passed through the Area Tree to the Renderer. The Renderer then renders the svg graphic as normal.

PDF Outline - This is output document extension. If rendering to pdf and this extensionis used then the bookmark information is passed to the pdf document. This information is then set on the document.

PDF Additions - This can be done with an XML Document extension. A simple xml document is defined that provides the appropriate information. When the document is rendered a handler converts the document into PDF markup.

For example:

```
<my:script-link script="app.execMenuItem('AcroSrch:Query');">
Search
</my:script-link>
```

to result in a text box referencing the following PDF action:

```
<<
/S /JavaScript
/JS (app.execMenuItem("AcroSrch:Query");)
>>
```

4 Status

4.1 To Do

4.2 Work In Progress

- mathml extension
- another xml -> svg extension
- svg text normal text if that can be handled otherwise stroked this is done automatically

4.3 Completed

- svg now in an xml handler, FOP can be used without batik
- bookmark extension improved a bit - changed bookmark extension, now requires a wrapping element bookmark