

# Upgrading from an Earlier Version of Apache FOP

\$Revision\$

## Table of contents

- 1 Important!.....2
- 2 What you need to know when you upgrade!.....2

## 1. Important!

If you're planning to upgrade to the latest FOP version there are a few very important things to consider:

- More than half of the codebase has been rewritten over the last three years. With version 0.93 the code has reached **production level**, and continues to improve with version 0.94.
- The API of FOP has changed considerably and is not backwards-compatible with versions 0.20.5 and 0.91beta. Version 0.92 introduced the **new stable API**.
- Since version 0.92 some deprecated methods which were part of the old API have been removed. If you upgrade from 0.91 beta, you will need to adjust your Java code. Similarly if you upgrade from 0.92 and use deprecated methods.
- If you are using a configuration file for version 0.20.5, you have to rebuild it in the new format. The format of the configuration files has changed since version 0.20.5. See `conf/fop.xconf` for an example configuration file. A XML Schema file can be found under `src/foschema/fop-configuration.xsd`.
- If you are using font metrics files for version 0.20.5 or 0.92 or earlier, you have to regenerate them in the new format. The new format is characterized by a version attribute on the top-level font-metrics element, whose value is 2.0. The absence of a version attribute will be interpreted as version 1.0, and such metrics files will no longer be parsed.
- The new code is much more strict about the interpretation of the XSL-FO 1.0 specification. Things that worked fine in version 0.20.5 might start to produce warnings or even errors now. FOP 0.20.5 contains many bugs which have been corrected in the new code.

### An example

While FOP 0.20.5 allowed you to have empty `fo:table-cell` elements, the new code will complain about that (unless relaxed validation is enabled) because the specification demands at least one block-level element (`{%block;}`)+, see [XSL-FO 1.0. 6.7.10](#) inside an `fo:table-cell` element.

- Extensions and Renderers written for version 0.20.5 will not work with the new code! The new FOP extension for [Barcode4J](#) is available since January 2007.
- The SVG Renderer and the MIF Handler have not been resurrected, yet! They are currently non-functional and hope for someone to step up and reimplement them.

## 2. What you need to know when you upgrade!

When you use your existing FO files or XML/XSL files which work fine with FOP version 0.20.5 against this FOP version some things may not work as expected. The following list will hopefully help you to identify and correct those problems.

- Check the [Compliance page](#) for the feature causing trouble. It may contain the necessary information to understand and resolve the problem.
- Not all 0.20.5 output formats are supported. PDF and Postscript should be fully supported. See [Output Targets](#) for a more complete description.
- As stated above empty table cells `<fo:table-cell></fo:table-cell>` are not allowed by the specification. The same applies to empty `static-content` and `block-container` elements, for example.
- 0.20.5 is not XSL-FO compliant with respect to sizing images (`external-graphic`) or `instream-foreign-object` objects. If images or SVGs are sized differently in your outputs with the new FOP version check [Bug 37136](#) as it contains some hints on what to do. The file "[examples/fo/basic/images.fo](#)" has a number of good examples that show the new, more correct behaviour.
- The `fox:outline` extension is not implemented in this version anymore. It has been superseded by the new bookmark elements from XSL-FO 1.1.
- The `fox:destination` extension is also not implemented in this version although it may be added in the future. See also [Bug 37157](#).