DocBook Installation for Ubuntu

Ubuntu 16.04, 18.04 and 20.04

Jeffrey Walton

Table of Contents

Introduction

This document is a guide to installing DocBook [https://docbook.org/whatis] on Ubuntu 16.04, 18.04 and 20.04. An installation on Fedora or CentOS will be similar but it is not the same due to different package names and different paths.

The installation requires eight components and it includes the DocBook stylesheets, XML and FO translators and processors, linters and checkers. `fop-hyphenate` is an additional but optional component.

The guide installs DockBook 4.5 and not version 5. If you want DockBook v5 then install `docbook5` and `docbook-xsl-ns` instead of `docbook` and `docbook-xsl`. You will also need to change some URI's to use version 5 paths.

Be careful when copying and pasting from this document. If you copy and paste commands from the document to the terminal then the dash used in commands may be translated into hyphens (em dash) used in typesetting. They are not the same thing in the terminal, and it usually leads to obscure failures when installing packages.

Components

DocBook requires a DTD, XSLTPROC, and FOP. In addition, Apache's FOP requires a Java subsystem. The components are as follows:

1. DocBook DTD – DocBook entity definitions

2. DocBook XSL – DocBook stylesheets
3. XMLLINT – XML validator and formatter
4. XSLTPROC – XML processor which reads the XML source and produces Formatted Objects (FOs)
5. Java – Runtime environment for Apache FOP
6. FOP – Apache's Formatted Object Processor which transforms FOs to output file types such as HTML, PS, and PDF
7. FOP-HYPH – used by FOP to hyphenate words in a DocBook
8. GhostScript – used to optimize a PDF produced by DocBook
9. QPDF – used to validate a PDF produced by DocBook or GhostScript

DockBook DTD

The DockBook DTD provides entity definitions. The DTD defines the structure, elements and attributes of an XML document. It is provided by the `docbook` package. To install the package on Ubuntu issue the following.

```
$ sudo apt-get install docbook
```

DockBook XSL

The DockBook XSL provides stylesheets. The stylesheets are used to transform and format the XML document for display. They are provided by the `docbook-xsl` and `docbook-xsl-doc-pdf` packages. To install the packages issue the following.

```
$ sudo apt-get install docbook-xsl docbook-xsl-doc-pdf
```

The stylesheets are installed in `/usr/share/.../fo/` directory. The important one is `docbook.xsl`.

```
$ ls /usr/share/xml/docbook/stylesheet/docbook-xsl/fo/
admon.xsl           division.xsl   info.xsl             refentry.xsl
annotations.xsl     docbook.xsl    inline.xsl           sections.xsl
autoidx-kimber.xsl  ebnf.xsl       keywords.xsl         spaces.xsl
autoidx-kosek.xsl   footnote.xsl   lists.xsl            synop.xsl
autoidx-ng.xsl      fop.xsl        math.xsl             table.xsl
...
```

XML lint

The XML lint provides validation and formatting of XML source files. It is provided by the `libxml2-utils` package. To install the package on Ubuntu issue the following.

```
$ sudo apt-get install libxml2-utils
```

After you install the program you should run `hash -r` to clear Bash's program cache.

XML processor

The XML processor translates XML source and produces Formatted Object (FO). It is provided by the `xsltproc` package. To install the package on Ubuntu issue the following.

```
$ sudo apt-get install xsltproc
```

After you install the program you should run `hash -r` to clear Bash's program cache.

Java JRE & JDK

The Java runtime enviroment is needed by Apache FOP. You can use the Runtime Environment (JRE) or the Development Kit (JDK). They are provided by the `openjdk-8-jre` or `openjdk-8-jdk` package on Ubuntu 16, and the `openjdk-11-jre` or `openjdk-11-jdk` package on Ubuntu 18. To install the package on Ubuntu issue the following.

```
$ sudo apt-get install openjdk-11-jdk
```

JRE and JDK versions change over time as distros move forward. To determine which JREs or JDKs are available on a Ubuntu system issue the following command.

```
$ apt-cache search '^openjdk' | grep -E '\-jre|\-jdk'
openjdk-11-jdk - OpenJDK Development Kit (JDK)
openjdk-11-jdk-headless - OpenJDK Development Kit (JDK)
openjdk-11-jre - OpenJDK Java runtime, using Hotspot JIT
openjdk-11-jre-headless - OpenJDK Java runtime, using Hotspot JIT
openjdk-11-jre-zero - Alternative JVM for OpenJDK, using Zero
...
```

After installing the JRE or JDK you can export `JAVA_HOME` from your login script. In the code below, `JAVA_HOME` is setup for Ubuntu 18. If you are using Ubuntu 16, then `JAVA_HOME` is `/usr/lib/jvm/java-8-openjdk-amd64`.

```
# .bashrc or .bash_profile
JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
export JAVA_HOME
```

After setting `JAVA_HOME` in the Bash startup file, `source` it to make the changes take effect now.

```
$ source ~/.bash_profile
$ echo $JAVA_HOME
/usr/lib/jvm/java-11-openjdk-amd64
```

If you wish to automate the setting of `JAVA_HOME`, then the following can be used in `.bashrc` or `.bash_profile`.

```
JAVA_HOME=$(readlink -f $(command -v java) | sed "s|/bin/java$||")
```

```
export JAVA_HOME
```

After you install Java you should run `hash -r` to clear Bash's program cache.

FO processor

The Formatted Object Processor (FOP) is an Apache binary. Ubuntu provides `fop` and `fop-doc` packages. You can download the Apache binary or install the FOP package.

To install the packages issue the following.

```
$ sudo apt-get install fop fop-doc
```

If you are happy with Ubuntu's version of FOP then you are finished. You can move on to the next section.

Alternatively, you can install the latest Apache FOP by downloading the binary from the Apache website. This can be useful if Ubuntu's version of FOP has trouble. To manually install FOP perform the following.

```
$ wget https://mirror.cogentco.com/pub/apache/xmlgraphics/
  fop/binaries/fop-2.8-bin.tar.gz
$ tar -xzf fop-2.8-bin.tar.gz
$ chmod +x fop-2.8/fop/fop
$ sudo mv fop-2.8 /usr/local/bin
```

Once `fop` is installed locally, add a wrapper script to execute it. In the `cat` below, `JAVA_HOME` is dynamically populated if the environmental variable is not set. If you wish, you can hard-code `/usr/lib/jvm/java-11-openjdk-amd64` for Ubuntu 18 or `/usr/lib/jvm/java-8-openjdk-amd64` for Ubuntu 16.

```
$ cat /usr/local/bin/fop
#!/usr/bin/env bash

if [[ -z "$JAVA_HOME" ]]; then
    JAVA_HOME=$(readlink -f $(command -v java) | sed "s|/bin/java$||")
    export JAVA_HOME
fi

CLASSPATH=/usr/local/bin/fop-2.8:/usr/local/bin/fop-2.8/lib
export CLASSPATH

/usr/local/bin/fop-2.8/fop/fop $*
```

After you install the processor you should run `hash -r` to clear Bash's program cache.

FO hyphenate

You can add hyphenation to the book by installing FOP XML Hyphenation Patterns. Download the *compiled binary* for FOP XML Hyphenation Patterns from the Objects For Formatting

Objects (OFFO) website [http://offo.sourceforge.net/]. Install `fop-hyph.jar` in FOP's `lib/` directory.

```
$ sudo cp fop-hyph.jar /usr/local/bin/fop-2.8/fop/lib/
$ ls /usr/local/bin/fop-2.8/fop/lib/*.jar
...
/usr/local/bin/fop-2.8/fop/lib/commons-io-2.11.0.jar
/usr/local/bin/fop-2.8/fop/lib/commons-logging-1.0.4.jar
/usr/local/bin/fop-2.8/fop/lib/fontbox-2.0.24.jar
/usr/local/bin/fop-2.8/fop/lib/fop-hyph.jar
/usr/local/bin/fop-2.8/fop/lib/serializer-2.7.2.jar
/usr/local/bin/fop-2.8/fop/lib/xml-apis-1.4.01.jar
/usr/local/bin/fop-2.8/fop/lib/xml-apis-ext-1.3.04.jar
...
```

GhostScript

The GhostScript package is used to optimize the PDF created by DocBook. To install the package on Ubuntu issue the following.

```
$ sudo apt-get install ghostscript
```

After you install the program you should run `hash -r` to clear Bash's program cache.

QPDF

The QPDF package is used to validate the PDF created by DocBook and GhostScript. To install the package on Ubuntu issue the following.

```
$ sudo apt-get install qpdf
```

After you install the program you should run `hash -r` to clear Bash's program cache.

Testing

You should test the installation using `command -v`. The Posix command is the standard way to check availability of a program.

```
$ command -v xmllint
/usr/bin/xmllint
$ command -v xsltproc
/usr/bin/xsltproc
$ command -v fop
/usr/local/bin/fop
```

You should also verify the version of `fop`.

```
$ fop -version
FOP Version 2.8
```

If the version is unexpected then verify the shell script located in `/usr/local/bin/fop`, and ensure it is executable. If the script is correct then run `hash -r` to clear the Bash program cache.

Fedora

If you install on Fedora then you should install the packages `docbook-dtds`, `docbook-style-xsl`, `docbook-utils` and `docbook-utils-pdf`.

You may need to manually tune the path to the DocBook stylesheets. If the book builds using `make-book.sh` then you are finished. If the book does not build then open the file `custom.xsl.in` and locate the line `<xsl:import href="!!DOCBOOK_XSL_FILE!!"/>`. Change `!!DOCBOOK_XSL_FILE!!` to the location of the stylesheet, like `/usr/share/sgml/docbook/xsl-stylesheets-1.79.2/fo/docbook.xsl`.

The new location will depend on the package you install. The location `/usr/share/sgml/...` is due to installing the Fedora package `docbook-style-xsl`, which is Norman Walsh's XSL stylesheets for DocBook.

Fedora also uses a `JAVA_HOME` path that changes over time. The current `JAVA_HOME` path for Fedora 32 is `/usr/lib/jvm/jre-11-openjdk-11.0.8.10-2.fc32.x86_64/`. Since `JAVA_HOME` changes over time, the `fop` script is more robust when it is crafted like shown below.

```
$ cat /usr/local/bin/fop
#!/usr/bin/env bash

if [[ -z "$JAVA_HOME" ]]; then
    JAVA_HOME=$(find /usr/lib/jvm -name 'java-11-openjdk-*' | head -n 1)
    export JAVA_HOME
fi

CLASSPATH=/usr/local/bin/fop-2.8:/usr/local/bin/fop-2.8/lib
export CLASSPATH

/usr/local/bin/fop-2.8/fop/fop $*
```

Running the script will set `JAVA_HOME` to the current version of the JVM.

```
$ find /usr/lib/jvm -name 'java-11-openjdk-*' | head -n 1
/usr/lib/jvm/java-11-openjdk-11.0.8.10-2.fc32.x86_64
```