MuPDF tools

Taco Hoekwater

Abstract

The application MuPDF (`http://mupdf.com`) is a very fast, portable, open-source PDF previewer and development toolkit actively supported by Artifex, the creators of Ghostscript (`http://artifex.com`).

But MuPDF is not *just* a very fast, portable, open-source PDF previewer and toolkit. It also comes with a handy collection of command-line tools that are easily overlooked.

The command-line tools allow you to annotate, edit, and convert documents to other formats such as HTML, SVG, PDF, and PNG. You can also write scripts to manipulate documents using JavaScript.

This small paper gives a quick overview of the possibilities.

1 Introduction

In recent versions of the MuPDF distribution, most of the tools have been combined into a single front-end program called `mutool`. This combines the functionality of the half dozen or so programs from earlier releases. If you use an older version of MuPDF, there will be little programs like `muclean`, but in the new combined version that functionality is now available as `mutool clean`. A similar command-line adjustment is needed for the other old command-line tool names.

In the following, I am using MuPDF 1.14.0. While the functionality of the separate tools remains roughly the same, not all versions of MuPDF have the exact same options. If you want to know the options that 'your' version of a command supports, just key in the name without arguments. Running `mutool` alone will provide a list of all known tools, and, e.g., `mutool clean` will show the list of options specific to the `clean` tool.

2 mutool clean — rewrite PDF file

If you are familiar with the general structure of PDF documents and you often work with PDF documents handed to you from other sources, this is probably the most valuable of all the tools.

Its main purpose is to 'clean up' a PDF. It can perform garbage collection on unused objects and clean up the page streams.

`mutool clean` can also convert a PDF into (near) ASCII by decompressing all the internal structures. The output is still a valid PDF, but since it has very little to no binary data any more, it can easily be inspected and possibly edited in a regular text editor. Beware though: this tends to make the file larger.

Alternatively, `mutool clean` can also convert a PDF into 'linearized' format for distribution on the web.

3 mutool convert — convert document

As the name suggests, this tool can convert a PDF file into a variety of different formats. Noteworthy supported formats in version 1.14 are: PNG, PNM, PCL, PS, PDF, SVG, HTML and plain text. Each of these has a number of sub-options to control the output format.

This is quite similar to the `convert` command from ImageMagick, except you do not need to have Ghostscript installed; it is generally faster, and uses less memory. On the down side, there are fewer output formats and options supported.

4 mutool create — create PDF document

With `mutool create`, you can create a PDF from text snippets that specify a page content stream. Each of these snippets becomes a page in the output PDF. It parses some special comments inside of those snippets to define images and fonts and page size, so for example a snippet could look like this:

```
%%MediaBox 0 0 300 300
%%Image Im0 /Users/taco/Downloads/22843.png
% Draw an image.
q
200 0 0 200 50 50 cm
/Im0 Do
Q
```

and the result would be a one-page PDF with that image centered in the page.

Because you have to write the page content stream, this is not a tool for beginners in PDF. Nevertheless, it is much easier to create a PDF this way than to write the PDF completely from scratch, because the required PDF objects and object references are generated by `mutool create`. Nevertheless, just using TeX is easier (albeit not as fast).

5 mutool draw — convert document

This is like `mutool convert`, except that it has more difficult to use options and uses a different syntax for those options. It is better to first see whether `mutool convert` can do what you want and only if it cannot, then look at `mutool draw`.

6 mutool trace — trace device calls

Produces a debug dump of the PDF document as an XML file. This can be useful to track what the

First published in *MAPS* 49 (2019.1), pp. 39–40. Reprinted with permission.

MuPDF library is actually doing, but too much information is lost from the PDF to do much else (at least, that is my experience so far).

7 mutool extract — extract font and image resources

Extracts images and embedded font resources from a PDF document, dumping them as separate files in the current directory.

8 mutool info — show information about PDF resources

Dumps detailed information about various PDF document internals to the standard output.

9 mutool merge — merge pages from multiple PDF sources into a new PDF

Combines one or more PDF documents (or pages from them) into a new combined PDF document. The fact that it can combine 'one' PDF means that this is an easy way to extract pages from a PDF. In fact, this is what I use to generate the separate article files for the on-line version of the Maps.

10 mutool pages — show information about PDF pages

In particular, `mutool pages` shows the various bounding boxes for all of the pages in XML format on the standard output.

11 mutool portfolio — manipulate PDF portfolios

PDF portfolios are a way of putting multiple independent PDFs into a single container PDF. With `mutool portfolio` you can create or modify the contents of such a portfolio.

12 mutool poster — split large page into many tiles

Splits each of the pages inside a PDF into tiles that then become the separate pages of the output PDF. It does not alter the page streams; `mutool poster` just creates adjusted bounding boxes for those separate pages so that they offer a different viewport to the original content stream.

This is a useful trick if you want to print a large PDF on a desktop printer, but it does not offer any extra features like registration marks.

13 mutool sign — manipulate PDF digital signatures

For now (this is a very new tool) this does nothing except verify an existing signature in a signed PDF document. And mine does not even do that yet, because it seems to be a build option that is not turned on in the MacOS version...

14 mutool run — run JavaScript

The MuPDF library has a quite elaborate interface to JavaScript that can be used with `mutool run` to execute scripts. These scripts have direct access to the command line and the MuPDF internals both for interpreting and for creating PDF documents, so very powerful things can be done.

The full interface is documented on the Artifex web pages.

15 mutool show — show internal PDF objects

`mutool show` is a tool for displaying the internal objects in a PDF file on the standard output. This can be very useful in combination with 'grep' for example, or if you do not want to load a multi-megabyte PDF in a text editor.

◇ Taco Hoekwater
`taco (at) bittext dot nl`