

Kurzpaper zum Workshop: PDF-KungFoo mit Ghostscript & Co.

Kurt Pfeifle <kurt.pfeifle@mykolab.com>

Chemnitzer Linux-Tage, März 2015-03

Vorbemerkung: Dieser Workshop wird kaum Folien verwenden. Stattdessen wird er fast alle Themen “live” vorführen. Zu seinen Zielsetzung gehören:

1. neueren Zuhörer(inne)n/Zuseher(inne)n eine eventuell vorhandene Scheu vor der Kommandozeile zu nehmen,
2. und ihnen Lust darauf zu machen, sich auf eigene Faust näher mit den Feinheiten der vorgestellten Tools befassen;
3. auch “erfahreneren” Nutzern einige neue Schmankerln zu bieten,
4. und ihnen eventuell bisher unbekannte Tools oder Kommandozeilen-Optionen vorzustellen.

Damit das klappt, ist es überhaupt nicht erforderlich, alles Gesehene sofort und an Ort+Stelle mitzumachen oder nachzuvollziehen. Denn das wird u.U. ziemlich schnell gehen, und man wird sich nicht alles sofort merken können. Deshalb stelle ich im Anschluß an den Workshop ein Tar-Archiv zur Verfügung. Darin wird eine ausführlichere Ausarbeitung des Workshops enthalten sein:

- alle verwendeten Kommandos,
- samt zugehörigen Erläuterungen,
- plus die verwendeten Beispiel-PDFs.

Auf diese Weise kann jede(r) Zuhörer(in) alles nachvollziehen, in aller Ruhe, zu Hause und im eigenen Tempo. Übrigens auch auf Windows-Systemen!

Nachfolgend eine kleine Vorschau in Stichworten, als Auswahl vorgeführter Kommandos. Erläuterungen zu den Kommandos gibt es auf der “Tonspur” im Workshop, die Visualisierungen parallel dazu auf der Leinwand.

Agenda-Überblick

Themenbereiche

- I. Teil 1: Kurz-Vorführung praktischer Anwendungen
- II. Teil 2: Ausführliche Vorstellung einiger Anwendungen
- III. Teil 3: Tipps + Tricks + Unbekanntes (Bonus-Track)

Kurz-Vorführung praktischer Anwendungen

- **Metadaten einer PDF**
- **In einer PDF enthaltene Fonts**
- **In einer PDF enthaltener Text**
- **In einer PDF enthaltene Bilder**
- **In einer PDF enthaltene JavaScripts**
- **In einer PDF enthaltene Objekte**

- In einer PDF enthaltene ‘Streams’
- In einer PDF enthaltene ‘verdächtige’ Objekte
- Schnell-Bleiche in PDF-Syntax
- Verwendet diese PDF Farben? Wenn ja, auf welchen Seiten?
- Ich möchte, dass man Text aus der PDF nicht so einfach extrahieren kann!

Vorschau auf die Workshop-Themen

```
## 'pdftotext'
pdftotext demo.pdf -
pdftotext -layout -f 3 -l 3 demo.pdf -
pdftotext -x .. -y .. -W .. -H .. -f 3 -l 3 demo.pdf -
pdftotext -layout incrupd.pdf

## 'pdfresurrect'
pdfresurrect incrupd.pdf

## 'pdfimages'
pdfimages -list demo.pdf
pdfimages -f 2 -l 2 -j demo.pdf extracted--

## 'pdftotext' (nochmals)
pdftotext -js demo.pdf

## 'pdf-parser.py'
pdf-parser.py -o 4 demo.pdf

## 'pdfid.py'
pdfid.py -e demo.pdf

## 'qpdf'
qpdf --qdf --object-streams=disable demo.pdf qdf.pdf
vim qdf.pdf

## 'mutool'
mutool clean demo.pdf
mutool extract demo.pdf

## 'ghostscript'
gs -o notext.pdf -sDEVICE=pdfwrite -dNoOutputFonts demo.pdf
gs -o - -q -sDEVICE=inkcov demo.pdf
gs -o metadata.pdf -sDEVICE=pdfwrite demo.pdf -c "showpage [/Keywords (keywords) /DOCINFO pdfmark"
```