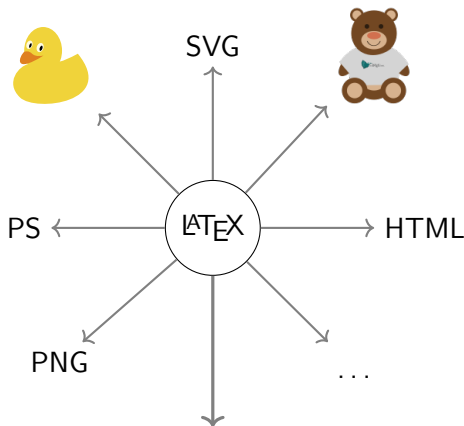


\LaTeX PDF management

Ulrike Fischer, Mönchengladbach
 \LaTeX Project Team

10.3.2021
Magdeburg–Mönchengladbach

\LaTeX kann unterschiedliche Formate erzeugen



Aber PDF ist derzeit das meist verwendete

Wenig PDF spezifisches direkt im Kernel

Einfache Dinge brauchen externe Pakete oder primitive Befehle:

Farben	color, xcolor	
Bilder	graphicx	
Seitengröße	diverse	<code>\pdfpageheight</code>
Links	hyperref	<code>\pdfstartlink</code>
PDF-version	hyperref	<code>\pdfminorversion</code>
Autor, Titel	hyperref	<code>\pdfinfo</code>
Seite drehen	pdflscape	<code>\pdfpageattr</code>

Problem 1: Konflikte

```
\pdfinfo{/Title (TitleA)} %Paket A  
\pdfinfo{/Title (TitleB)} %Paket B  
\pdfcatalog{/Lang (en-US)}  
\pdfcatalog{/Lang (de-DE)}
```

⇒ PDF Info: /Title (TitleA) /Title (TitleB)
⇒ PDF Catalog: /Lang (en-US) /Lang (de-DE)

- Nicht unbedingt fatal, aber falsch, nicht eindeutig und nicht valide.

Viele Wege führen zu PDF ...

- pdf \LaTeX
- lua \LaTeX
- xe \LaTeX
- \LaTeX + dvipdfmx
- dvlua \LaTeX + dvipdfmx
- \LaTeX + dvips + ps2pdf
- \LaTeX + dvips + distiller

... und die anderen Ausgabeformate sind auch noch da

Problem 2: Backend Unterstützung

... und fast jeder braucht andere Befehle:

```
pdfLATEX    \pdfinfo  
luaLATEX    \pdfextension info  
XeLATEX    \special{docinfo <<...>>}  
dvips       \special{ps: mark /#1 ~ #2 /DOCINFO pdfmark}  
...
```

Problem 2: Backend Unterstützung – viele Treiber

... und alle Pakete müssen dafür Code entwickeln und warten:

hyperref htex4ht.def, hypertex.def, hdvipdfm.def, htexture.def,
hdvips.def, hvtext.def, hdvipson.def, hvtexthtml.def,
hdviwind.def, hvtextmrk.def, hluatex.def, hxetex.def,
hpdfptex.def, pdfmark.def

bookmark bkm-dvipdfm.def, bkm-vtext.def, bkm-pdfptex.def,
bkm-dvips.def

graphics pdfptex.def, luatex.def, xetex.def, dvipdfmx.def ...

attachfile2 atfi-pdfptex.def, atfi-luatex.def, atfi-dvips.def ...

pdfbase (im Code)

...

Na und?

Es geht doch seit 30 Jahren!

na-und.png

„I remember when you could only lose a chess game to a supercomputer.“

- Drucken ist nicht mehr das Hauptziel
- PDF Standards werden wichtiger
- Barrierefreiheit wird wichtiger
- Exporte in andere Formate werden wichtiger
- Metadaten werden wichtiger

⇒ L^AT_EX Projekt „Tagged PDF“: Kooperationen mit PDF Association und Adobe

⇒ *viel* mehr PDF spezifischer Code

⇒ bessere Schnittstellen nötig

- ❶ Konflikte bei zentralen Befehlen auflösen
- ❷ bessere und verlässliche Unterstützung der Backends
- ❸ Neue Hooks und neue Schnittstellen wo nötig einfügen

`\pdfmanagement_add:nnn`

ersetzt

- `\pdfinfo`
- `\pdfcatalog`
- `\pdfpageattr`
- `\pdfpagesattr`
- `\pdfpageresources`

Ziel 1: Konflikte auflösen

```
\pdfinfo{/Title (TitleA)}  
\pdfinfo{/Title (TitleB)}  
\pdfcatalog{/Lang (en-US)}  
\pdfcatalog{/Lang (de-DE)}
```

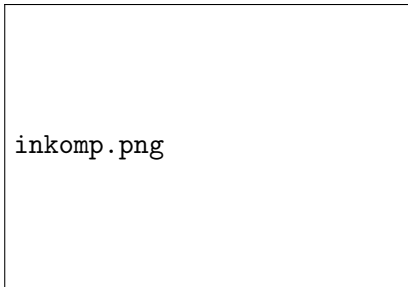
```
\pdfmanagement_add:nnn{Info} {Title} {(TitleA)}  
\pdfmanagement_add:nnn{Info} {Title} {(TitleB)}  
\pdfmanagement_add:nnn{Catalog} {Lang} {(en-US)}  
\pdfmanagement_add:nnn{Catalog} {Lang} {(de-DE)}
```

⇒ /Title (TitleB) und /Lang (de-De)

Alle Pakete mit

`\pdfinfo,`
`\pdfcatalog,`
`\pdfpageattr,`
`\pdfpagesattr,`
`\pdfpageresources`

sind nicht kompatibel!




*„There can be no peace until they renounce their
Rabbit God and accept out Duck God.“*

z.B.

hyperref-Treiber, pgf, hyperxmp, attachfile, pdfscape, ocgx2, ...

- pdfscape wird geändert \Rightarrow pdfpages muss auch angepasst werden
- hyperref wird geändert \Rightarrow beamer muss sich anpassen
- ...

```
\RequirePackage  
  {pdfmanagement-testphase}  
%Triggerbefehl  
\DeclareDocumentMetadata{...}  
\documentclass{...}
```

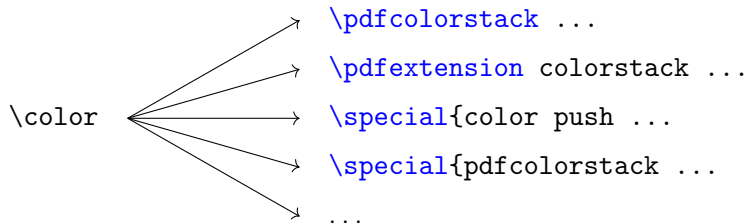


sicherheit.png

Ziel 2: bessere Backendunterstützung

Abstraktion

Backendbefehle



Mehr Backendbefehle direkt in \LaTeX

- `l3backend`

File: `l3backend-pdftex.def` 2021-03-02 L3 backend
support: PDF output (pdfTeX)

- `l3backend-testphase`

File: `l3backend-testphase-pdftex.def` 2021-03-07
LaTeX PDF management testphase

- Option:

`\DeclareDocumentMetadata{backend=dvipdfmx}`

Ziel 2: bessere Backendunterstützung

Neue Module für Abstraktionen

l3pdf, l3color, l3draw

l3pdfannot, l3pdfxform, l3pdffile, l3pdffield

```
\pdf_version_gset:n
```

```
\color_select:nn{rgb}{1,0,0}
```

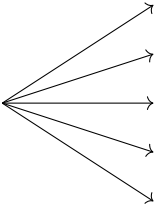
```
\pdf_object_new:nn
```

```
\pdfannot_link:nnn
```

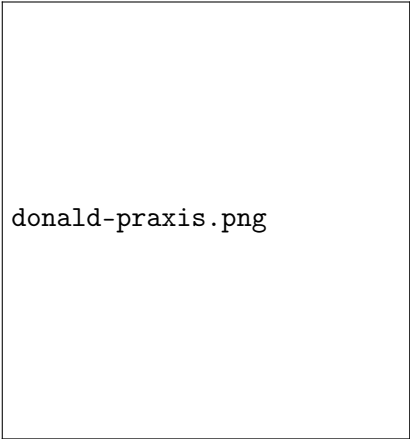
```
\pdffield_checkbox:n
```

Ziel 2: bessere Backendunterstützung

Abstraktionen sind nicht trivial

Abstraktion	Backendbefehle
<code>\pdf_version_gset:n</code>	
	✓ pdf \LaTeX
	✓ lua \LaTeX
	✓ (x)dvipdfmx
	???? dvips
	— dvisvgm

Was bringt es?



donald-praxis.png

Was bringt es? Einfacherer Code

Beispiel: transparent

👉 transparent.sty \Rightarrow transparent-ltx.sty

Beispiel: pdfscape

👉 pdfscape.sty \Rightarrow pdfscape-ltx.sty

Was bringt es? Mehr Funktionen

Beispiel: Seite drehen 🖱 rotate.tex, rotate.pdf


Was bringt es? Hyperref

Statt

~~hpdftex.def~~
~~hluatex.def~~
~~hxetex.def~~
~~hdvips.def~~
~~pdfmark.def~~

⇒

hgeneric-testphase.def

- ⇒ Neues muss nur einmal eingebaut werden
- ⇒ Fehler müssen nur einmal korrigiert werden
 - andere Farben
 - mehr Möglichkeiten Linkeigenschaften zu ändern
 - neue Checkboxes  checkbox.pdf

Was bringt es? Container für Dokumentdaten

`\AddToDocumentProperties`

`\ShowDocumentProperties`

`\GetDocumentProperties`

 `documentproperties.tex`

Was bringt es? pdf/A-2b + Farbprofile

```
\usepackage{hyperxmp}
\usepackage[pdfa]{hyperref}
\hypersetup{...}
\immediate\pdfobj stream attr{/N 3} file{sRGB.icc}
\pdfcatalog{%
  /OutputIntents [
    <<
      /Type /OutputIntent
      /S /GTS_PDFA1
      /DestOutputProfile \the\pdflastobj\space 0 R
      /OutputConditionIdentifier (sRGB)
      /Info (sRGB)
    >>
  ]}
```

Was bringt es? pdf/A-2b + Farbprofile

```
\ExplSyntaxOn
\pdf_object_unnamed_write{fstream}{{sRGB.icc}{{/N~3}}}
\pdfmanagement_add:nx{Catalog}{OutputIntents}
{
  <<
    /Type /OutputIntent
    /S /GTS_PDFA1
    /DestOutputProfile~\pdf_object_ref_last:
    /OutputConditionIdentifier (sRGB)
    /Info (sRGB)
  >>
}
\ExplSyntaxOff
```

Was bringt es? pdf/A-2b + Farbprofile

```
\RequirePackage{pdfmanagement-testphase}  
\DeclareDocumentMetadata{pdfstandard=A-2b}  
\documentclass{...}  
\usepackage{hyperxmp}  
\usepackage{hyperref}
```

Was bringt es? newpax: Wiedereinfügen von Links

Basiert auf pax von Heiko Oberdiek

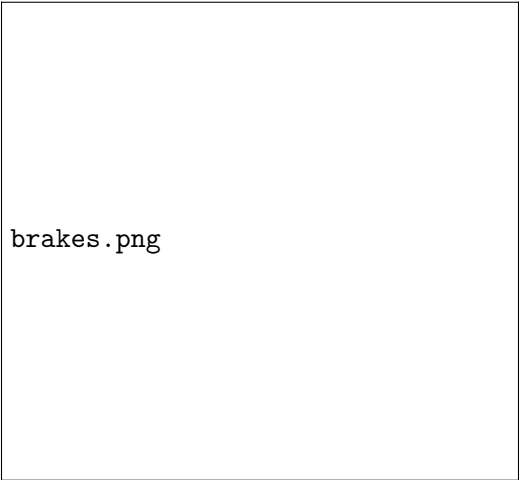
Beispiel:  newpax-test.tex, newpax-test.pdf

Ausblick: Tester und feedback benötigt!

- alle Arten von Dokumenten
- diverse Ausgabeformaten und Workflows – wir wollen ja nicht svg oder tex4ht kaputtmachen ...

Ausblick: Und wann ist es fertig?

Es wird noch was dauern . . .



brakes.png

moral of the story:
never read the documentation, bad things happen
David Carlisle

Publications by topic: PDF, Tagging, Accessibility:

<https://www.latex-project.org/publications/indexbytopic/pdf/>

```
texdoc -l pdfmanagement
```

```
1 ... \pdfmanagement-testphase\pdfmanagement-testphase.pdf
  = Package documentation
2 ... \pdfmanagement-testphase\hyperref-generic.pdf
3 ... \pdfmanagement-testphase\l3backend-testphase.pdf
4 ... \pdfmanagement-testphase\l3pdfannot.pdf
5 ... \pdfmanagement-testphase\l3pdfdict.pdf
6 ... \pdfmanagement-testphase\l3pdfffield.pdf
7 ... \pdfmanagement-testphase\l3pdfffile.pdf
8 ... \pdfmanagement-testphase\l3pdfmanagement.pdf
9 ... \pdfmanagement-testphase\l3pdfmeta.pdf
10 ... \pdfmanagement-testphase\l3pdfftools.pdf
11 ... \pdfmanagement-testphase\l3pdfxform.pdf
12 ... \pdfmanagement-testphase\ltdocinit.pdf
13 ... \pdfmanagement-testphase\pdfmanagement-firstaid.pdf
```