

LaTeX, beamer, tikz und Co.

91. Technische Aspekte

Thomas Worsch

Fakultät für Informatik
Karlsruher Institut für Technologie

Wintersemester 2017/2018

`\PassOptionsToPackage`

Reihenfolge für das Laden von Paketen

`hyperref`

Paket `pkgloader`

Paket `mylatexformat`

Überblick

\PassOptionsToPackage

Reihenfolge für das Laden von Paketen

Paket mylatexformat

\PassOptionsToPackage

- ▶ `\PassOptionsToPackage{⟨option list⟩}{⟨pkg name⟩}`
- ▶ *darf auch vor `\documentclass` stehen*
- ▶ sorgt dafür, dass bei einem späteren
`\usepackage [⟨options⟩] {⟨pkg name⟩}`
«das Paket glaubt», dass die Optionen aus `⟨option list⟩`
«mit angegeben seien»

\PassOptionsToPackage

- ▶ `\PassOptionsToPackage{⟨option list⟩}{⟨pkg name⟩}`
- ▶ *darf auch vor `\documentclass` stehen*
- ▶ sorgt dafür, dass bei einem späteren `\usepackage [⟨options⟩] {⟨pkg name⟩}` «das Paket glaubt», dass die Optionen aus `⟨option list⟩` «mit angegeben seien»
- ▶ nützlich, wenn man `⟨pkg name⟩` gar nicht selbst lädt, sondern intern von einem anderen Paket geladen wird

Überblick

`\PassOptionsToPackage`

Reihenfolge für das Laden von Paketen

`hyperref`

Paket `pkgloader`

Paket `mylatexformat`

Manchmal ist die Reihenfolge wichtig

- ▶ aber leider nicht immer klar dokumentiert
 - ▶ u. U. nicht mal dokumentierbar
- ▶ Standardbeispiele:
 - ▶ `hyperref`, `cleveref`
 - ▶ `amsmath`, `ntheorem`

hyperref/cleveref

- ▶ `\usepackage{hyperref}`
«Make sure it comes *last* of your loaded packages, to give it a fighting chance of not being over-written, since its job is to redefine many \LaTeX commands.»
- ▶ `\usepackage{cleveref}`
«Basically, cleveref must be loaded *last*.»
- ▶ mehr Informationen in <http://mirrors.ctan.org/macros/latex/contrib/hyperref/README.pdf>
- ▶ <http://tex.stackexchange.com/questions/1863/which-packages-should-be-loaded-after-hyperref-instead-of-before>

Pakete automatisch in der «richtigen» Reihenfolge laden

einfache Variante

- ▶ Paket `pkgloader` vor `\documentclass{<class>}` laden

```
\RequirePackage{pkgloader}  
\documentclass{article}  
...  
\begin{document}  
...  
\end{document}
```

Pakete automatisch in der «richtigen» Reihenfolge laden

flexible Variante

```
\RequirePackage{pkgloader}
\documentclass{article}
\usepackage{algorithm}
\usepackage{hyperref}
\usepackage{float}
\LoadPackagesNow    % ab hier alles "normal"
...
\begin{document}
...
\end{document}
```

Überblick

`\PassOptionsToPackage`

Reihenfolge für das Laden von Paketen

Paket `mylatexformat`

Was ist ein Format?

- ▶ ein $\text{T}_\text{E}\text{X}$ -Format ist eine Datei, die
 - ▶ schnell geladen werden kann
 - ▶ zusätzliche Definitionen von Kommandos, Umgebungen, etc. enthält
- ▶ *«Format files are binary memory dumps that are closely tied to the engine they have been created with»*
 - ▶ `tex` und `pdftex` «verstehen» nur verschiedene Formate
- ▶ <http://tex.stackexchange.com/questions/324559/guidelines-for-using-mylatexformat-with-luatex>
 - ▶ *«you can not dump lua state or opentype fonts so you need to avoid any luatex-specific setup»*
 - ▶ *«So, no `polyglossia`, no `fontspec`, no `tikz`, no `pgfplots`, etc.»*

Eigene Formate mit `mylatexformat`

- ▶ «to make a format based on the preamble of any \LaTeX file»
- ▶ dokumentierte Ausnahme:
 - ▶ `minitoc`
- ▶ prinzipielles Vorgehen
 1. Erzeugen des Formats
 - ▶ mit `mylatexformat`
 - ▶ siehe <http://liinwww.ira.uka.de/~thw/vl-latex-co/t-mylatexformat.zip>
 2. Benutzen des Formats
 - ▶ wie mit allen Formaten

mylatexformat — Erzeugung einer Format-Datei

- ▶ Ausgangspunkt: ein \LaTeX -Dokument `\langle filename \rangle.tex`
- ▶ Präambel soll
 - ▶ ganz bzw.
 - ▶ bis zur Zeile `\csname endofdump\endcsname`in ein Format «gedumpt» werden

- ▶

```
etex -ini -interaction=nonstopmode \  
-jobname="\langle format-name \rangle" \  
"&pdflatex" mylatexformat.ltx \  
"""\langle filename \rangle.tex"""
```
- ▶ erzeugt Formatdatei `\langle format-name \rangle.fmt`

Benutzung einer eigenen Format-Datei

- ▶ `etex -fmt <format-name> ...` oder
- ▶ in der ersten Zeile des \LaTeX -Dokuments
`%&<format-name>`
und Aufruf von \LaTeX mit Option
`-parse-first-line` (o.ä.)