

LaTeX im Studium

# Fortgeschrittene Präsentationen mit LaTeX

Jörn Clausen

joern@TechFak.Uni-Bielefeld.DE

# Übersicht

- Präsentationen mit seminar

# Übersicht

- Präsentationen mit seminar
- Effekte mit PPower4

# Präsentationen

- Lösungen für  $\LaTeX$ :  $\text{SL}\text{I}\text{T}\text{E}\text{X}$ , slides,  $\text{Foil}\text{E}\text{X}$ , seminar, ...
- alle brauchbar
- PDF-Datei im Fullscreen-Modus anzeigen
- bereits im letzten  $\LaTeX$ -Kurs:
  - erweiterte Folien-Umgebung
  - Seitengestaltung

# Aufgaben

- Übersetze die Datei `presentation.tex` mit Hilfe des Makefiles. Sieh Dir das Ergebnis mit Ghostview an. Was fällt auf?
- Die Seiten der PostScript-Datei können mit der Anweisung  

```
pstops '0U(1w,1h)' infile.ps outfile.ps
```

gedreht werden. Baue die Anweisung auf sinnvolle Weise in das Makefile ein.
- Binde die Pakete `times` bzw. `syntax` ein (nacheinander, jeweils nur eins) und sieh Dir das Ergebnis an.

# Aufgaben

- Übersetze die Folien-Datei mit pdfL<sup>A</sup>T<sub>E</sub>X. Füge eine entsprechende Regel in das Makefile ein.

Welche Probleme treten dabei auf? Wie sieht das Ergebnis im Acrobat Reader aus?

- Binde das Paket `fixseminar` ein und übersetze die Datei erneut. Was passiert nun?
- Erzeuge aus der veränderten Quell-Datei erneut die PostScript-Variante. Hat sie sich verändert?

# PPower4

- funktioniert mit herkömmlichen Folien-Stilen
- benötigt pdf $\LaTeX$
- Post-Prozessor `ppower4` in Java:
  - erkennt Markierungen in PDF-Datei
  - teilt Seite an Markierungen
- Vorteil: Folie wird nur einmal formatiert, keine Darstellungsfehler
- farbiger Folienhintergrund
- `pdfthumb` als Zugabe

# die orange \pause

- `pause.sty`
- `pdflatex + ppower4`

```
\begin{itemize}
\item foo\pause
\item bar\pause
\item baz\pause
\item boom
\end{itemize}
```



- foo

- Überblendeffekte: `\pauseDissolve`, `\pauseHBlinds`, ...



# die orange \pause

- `pause.sty`
- `pdflatex + ppower4`

```
\begin{itemize}
\item foo\pause
\item bar\pause
\item baz\pause
\item boom
\end{itemize}
```



- foo
- bar

- Überblendeffekte: `\pauseDissolve`, `\pauseHBlinds`, ...

# die orange \pause

- `pause.sty`
- `pdflatex + ppower4`

```
\begin{itemize}
\item foo\pause
\item bar\pause
\item baz\pause
\item boom
\end{itemize}
```



- foo
- bar
- baz

- Überblendeffekte: `\pauseDissolve`, `\pauseHBlinds`, ...

# die orange \pause

- `pause.sty`
- `pdflatex + ppower4`

```
\begin{itemize}
\item foo\pause
\item bar\pause
\item baz\pause
\item boom
\end{itemize}
```



- foo
- bar
- baz
- boom

- Überblendeffekte: `\pauseDissolve`, `\pauseHBlinds`, ...

# Aufgaben

- Binde den `pause`-Stil ein und füge einige `\pause`-Anweisungen in die Disney-Folie ein. Sieh Dir die entstehende PDF-Datei an.

- Nach der Übersetzung mit `pdfLATEX` muß die PDF-Datei mit `ppower4` weiterverarbeitet werden:

```
$ ppower4 presentation.pdf presentation-screen.pdf
```

Füge eine entsprechende Regel in das Makefile ein.

- Welches Problem gibt es mit dem Aufbau der Folien?
- Probiere einige der alternativen Überblendeffekte aus:

```
\pauseDissolve, \pauseHBlinds, \pauseVBlinds,  
\pauseHOSplit, \pauseHISplit, \pauseWipe{90},  
\pauseGlitter{315}
```

# Seitenaufbau steuern

- Seite wird durch `\p` in Regionen unterteilt
- jede Region hat „pause level“
- normalerweise fortlaufende Zuweisung der level
- explizite Zuweisung mit `\p{...}`
- sinnvollerweise direkt nach `\p` verwenden

# Aufgaben

- Ergänze jede Disney-Figur um eine `\pause`-Anweisung. Füge unmittelbar nach der 4. `\pause`-Anweisung (Donald) den Befehl

```
\pauselevel{=1}
```

ein. Wie wird die Liste jetzt aufgebaut?

- Relative Sprünge können durch Voranstellen eines Plus- bzw. Minuszeichens erzeugt werden:

```
\pauselevel{=-2}
```

In welcher Reihenfolge werden die Punkte nun eingeblendet?

- Sorge dafür, daß, nachdem die Punkte 3 und 5 gleichzeitig eingeblendet wurden, es in der richtigen Reihenfolge weitergeht.
- Was ist zu tun, damit die Fußzeile von Anfang an auf den Folien zu sehen ist?

# Aufgaben

- Entferne die `\pauselevel`-Anweisungen wieder. Füge vor dem ersten Listeneintrag die Anweisung

```
\pauselevel{=8 -1}
```

ein. Wie wird die Liste nun aufgebaut?

- Durch die Anweisungen

```
\pauselevel{:5}
```

```
\pauselevel{:+1}
```

kann man angeben, bis zu welchem Level eine Region dargestellt werden soll. Auf folgenden Folien wird sie nicht mehr angezeigt. Sorge dafür, daß Donald wieder verschwindet, sobald sein Onkel Scrooge erscheint.

# Aufgaben

- Um eine Region ab einer bestimmten Folie ein- und nach einer anderen Folie wieder auszublenden, kann man die Notationen kombinieren:

```
\pauselevel{=3 :5}
```

- Mit dem T<sub>E</sub>X-Befehl `\rlap` wird ein Text gesetzt und die Schreibmarke anschließend wieder zurückgeschoben:

```
Donald is their \rlap{father}uncle
```

Donald is their **father**

- Füge geeignete `\pause`- und `\pauselevel`-Anweisungen in den obigen Satz ein, so daß zunächst das Wort „father“ gesetzt wird, um auf der nächsten Folie durch „uncle“ ersetzt zu werden.



# Farben

- Farben durch Paket `color`

- neue Farben definieren:

```
\definecolor{unigruen}{rgb}{0.00,0.85,0.09}  
\textcolor{unigruen}{Universit"at Bielefeld}
```

Universität Bielefeld

- andere Farbmodelle:

```
\definecolor{medgray}{gray}{0.5}  
\definecolor{darkmagenta}{cmyk}{0,1,0,0.2}  
\textcolor{medgray}{gr"auliche}  
\textcolor{darkmagenta}{Telekomiker}
```

gräuliche Telekomiker

- Hintergrundfarben mit PPower4: Paket `background`

# Aufgaben

- Binde die Pakete `color` und `background` ein. Definiere folgende Farben:

```
\definecolor{rgbwhite}{rgb}{1,1,1}
```

```
\definecolor{lightblue}{rgb}{0.7,0.7,1}
```

- Färbe den Hintergrund der Folien gleichmäßig ein:

```
\pagecolor{lightblue}
```

- Erzeuge einen Farbverlauf mit

```
\vpagecolor[lightblue]{rgbwhite}
```

Analog gibt es den Befehl `\hpagecolor`.

- Ändere die Textfarbe mit dem Befehl `\color{...}`.
- Finde geeignete Farben, die die Lesbarkeit nicht einschränken...