

Einführung in das Textsatzsystem



L^AT_EX



07 – Gleitumgebungen II

ARTISTIC LION

13. Dezember 2013

Inhalt

- 1 Graphikarten
- 2 Einbinden externe Graphiken
- 3 Paket graphicx
- 4 Zeichenpakete

Bilder

Pixelgraphik

- eine Menge an Punkten
- jedem Punkt wird eine Farbe zugeordnet
- Ergebnis von Photos, Scans, etc.
- keine Skalierbarkeit

Vektorgraphik

- Beschreibung durch mathematische Objekte
- Kurven (Bézier-Kurven, Polynome, ...) o. ä.
- beliebige Skalierbarkeit
- meist kleine Dateigröße
- moderne Schriften sind Vektorgraphiken!

T_EX und Bilder

- T_EX stammt aus einer Zeit, in der Texte den Informationsaustausch dominierten
 - T_EX kennt *keine* Möglichkeit, externe Bilder einzubinden
 - T_EX kennt nur Boxen
 - Für alles andere gibt es `\special`
 - schreibt Informationen für den Ausgabetreiber
- ⇒ abhängig vom „Ausgabegerät“!

externe Graphiken

- L^AT_EX bietet Möglichkeit, extern erzeugte Bilder einzubinden:
- **`\includegraphics[optionen]{bildname}`**

externe Graphiken

- L^AT_EX bietet Möglichkeit, extern erzeugte Bilder einzubinden:

- **`\includegraphics[optionen]{bildname}`**

- Gleitumgebung `figure`

- je nach Treiber sind verschiedene Formate möglich,
u. a. jpeg, ps, pdf, gif, tiff, ...

⇒ unter Umständen Umformatierung nötig!

- LuaL^AT_EX bietet *native* Unterstützung für png, jpg und pdf (*nicht* svg und (e)ps)
- ps/eps werden auf neuen Systemen *automatisch* on-the-fly umgewandelt!
- für ältere Systeme: Umwandlung nach pdf mit Paket `epstopdf`, erfordert evtl. auf älteren Systemen `\write18` (`--shell-escape`)

externe Graphiken

Vorteile externer Graphiken

- freie Gestaltungsmöglichkeit
- Erzeugung in WYSIWYG-Graphikprogrammen
- Unabhängigkeit vom Dokument
- spezialisierte Programme für jeden Zweck

externe Graphiken

Vorteile externer Graphiken

- freie Gestaltungsmöglichkeit
- Erzeugung in WYSIWYG-Graphikprogrammen
- Unabhängigkeit vom Dokument
- spezialisierte Programme für jeden Zweck

Nachteile externer Graphiken

- getrennt vom Dokument \Rightarrow Portabilität leidet
- Layout passt nicht zum Schriftbild
- Bildbeschriftungen müssen zur Browserschrift oder Matheschrift passen
- Treiberabhängigkeit

Inkompatible Formate

- falls eine Graphik benötigt wird, mit welcher der Treiber nicht umgehen kann:

```
\Declaregraphicsrule{<Endung>}{<Typ>}{<Größe>}{<Befehl>}  
\DeclareGraphicsRule{.jpg}{eps}{}{'jpg2eps #1'}
```

- externe Umwandlung empfohlen:
z. B. IrfanView, gimp ...
- T_EX live liefert auch einige Umwandlungstools mit (pstopdf, pstops, ...)

graphicx

- Grundbefehl: `\includegraphics[optionen]{datei}`
 - Dateiendung muss nicht angegeben werden
 - keine absoluten Pfadangaben verwenden! (Portabilität)
- ⇒ statt `\includegraphics{/home/arno/Arbeit/Bilder/name}`
besser `\includegraphics{name}`
oder `\includegraphics{Bilder/name}`

graphicx

- Grundbefehl: `\includegraphics[optionen]{datei}`
 - Dateiendung muss nicht angegeben werden
 - keine absoluten Pfadangaben verwenden! (Portabilität)
- ⇒ statt `\includegraphics{/home/arno/Arbeit/Bilder/name}`
besser `\includegraphics{name}`
oder `\includegraphics{Bilder/name}`
- noch besser: `\graphicspath{{Photos}{Plots}{Scans}}`
 - erwartet eine *Liste* an geklammerten Pfaden, es muss also `\graphicspath{{Bilder}}` doppelt geklammert werden!

graphicx

- Paket `graphicx` erweitert `graphics` um ein key-value Interface, z. B.
`\includegraphics[scale=.5]{bild}`

Einbinden von Graphiken

```
\includegraphics[width=2cm]{500}  
\includegraphics[width=.3\textwidth,angle=25]{500}
```





Optionen für includegraphics

\includegraphics kennt viele Optionen, z. B.

Schlüssel	Werte
scale	0.8
width	.2\textwidth
height	2em
keepaspectratio	true oder false
angle	50
bb	0 0 10 20
clip	true oder false

⇒ siehe Dokumentation zu graphicx

Teilbilder – subfig

- ermöglicht mehrere Einzelobjekte in einem Gleitobjekt
- Paket `subfig` definiert `\subfloat[]{} (!)`
- optionales Argument ist (Einzel-)Beschriftung
- obligatorisches Argument ist Inhalt
- inkompatibel mit `beamer` ...

```
\begin{table}  
\subfloat[Erste Tabelle]{  
\begin{tabular}{ccc} a & b & c \end{tabular}  
}  
\subfloat[Zweite Tabelle]{  
\begin{tabular}{ccc} a & b & c \end{tabular}  
}  
\caption{Zwei Tabellen}  
\end{table}
```


Teilbilder – subfloat

- ermöglicht unabhängige Platzierung von Teilobjekten (1.a, 1.b, ...)
- Legenden einzeln, aber zusammenhängend
- Umgebungen subfigures und subtables

Teilbilder – subfloat

```
\begin{subfigures}  
\begin{figure}  
\fbox{Bild 1}\caption{Legende 1}  
\end{figure}  
\begin{figure}  
\fbox{Bild 2}\caption{Legende 2}  
\end{figure}  
\end{subfigures}
```

Bild 1

Abbildung: Legende 1

Bild 2

Abbildung: Legende 2

Legenden seitlich setzen

- Paket `sidescap` ermöglicht Satz von Legenden *neben* Objekten
- Umgebungen `SCfigure` und `SCtable`
- viele Optionen zur Formatierung der Legende
- Mit KOMA-Skript auch möglich!

sidecap

```
\begin{SCtable}[0.5][t]  
\fbox{Eine Tabelle}  
\caption{Eine Legende neben der tollen Tabelle}  
\end{SCtable}
```

Bilder im Text

- aus Textverarbeitungssystemen bekannt:
Text, der Bild umfließt (nicht rechteckig, sondern der Form angepasst)
 - typographisch fragwürdig – Abhebung des Bildes vom Text
 - Umfließen stört Lesefluss erheblich
 - T_EX kann prinzipiell keine Graphiken umfließen
 - mit größerem Aufwand möglich \Rightarrow Paket `cutwin` (Internetsuche nach `dtk cutwin` liefert die DTK-Ausgabe mit einem Artikel zu `cutwin`)
 - Platzierung am Rand einfach möglich
- \Rightarrow Pakete `wrapfig`, `picins`, `picinpar`, `floatflt`

wrapfig

```
\blindtext
\begin{wrapfigure}{l}[0.1\width]{3cm}
\includegraphics[width=1cm]{unilogo}
\end{wrapfigure}
\blindtext[3]
```

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift -- mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift -- mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift -- mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.



wrapfig

`wrapfig` bietet folgende Optionen:

```
\begin{wrapfigure}[zeilen]{position}[überhang]{breite}
```

<code>zeilen</code>	Anzahl schmaler Zeilen
---------------------	------------------------

<code>position</code>	Seite, auf die gesetzt wird.
-----------------------	------------------------------

<code>L,R</code>	links bzw. rechts, gleitend
------------------	-----------------------------

<code>l,r</code>	links bzw. rechts, nicht gleitend
------------------	-----------------------------------

<code>i,o,I,O</code>	innen bzw. außen, analog
----------------------	--------------------------

<code>überhang</code>	Einrückung in den Rand
-----------------------	------------------------

<code>breite</code>	Breite der Abbildung
---------------------	----------------------

wrapfig

- wrapfig bietet auch wraptable
- wrapfigure und wraptable unterstützen `\caption` und `\label` :
- Verwendung wie normale Gleitumgebungen
- `\width` für natürliche Breite

picinpar

Löcher im Absatz

- `picinpar` ermöglicht Satz von beliebigem Inhalt innerhalb eines Absatzes
- Umgebung `window`: keine weitere Formatierung
- Umgebungen `figwindow` und `tabwindow` ermöglichen konsistenten Satz von Unterschriften
- **Vorsicht:** evtl. Probleme mit Gleitumgebungen (`figure`, `table`)

floatflt

- `floatflt` kann Bilder am Anfang eines Absatzes setzen
- auf CTAN zu finden, nicht bei T_EXlive dabei

```
\begin{floatingfigure}{60mm}
  \includegraphics[width=60mm]{B-7-Physik-Kloster.jpg}
  %
  \caption[Grundriß des zweiten Obergeschosses im
Dominikanerkloster \newline Heidelberger Jahrbücher
der Literatur 13,4 (1820)]
  {\label{Physik-Kloster} Grundriß des zweiten
Obergeschosses im Dominikanerkloster}
\end{floatingfigure}
```

Zeichenpakete

- PS-Tricks und TikZ
(PostScript-Tricks, TikZ ist kein *Zeichenprogramm*)
- bieten immens große Möglichkeit, Graphiken zu erstellen
- viele spezielle Erweiterungspakete, vor allem zu PS-Tricks
- z. B. Erstellen von Knotendiagrammen, Schaltplänen, u. v. m.

pst-circ

```
\begin{pspicture}(3,2)
\pnode(0,1){A}\pnode(3,1){B}
\pnode(3,0){C}\pnode(0,0){D}
\resistor(A)(B){$R$}
\capacitor(B)(C){$C$}
\LED(C)(D){$\mathcal{D}$}
```

pst-circ

```
\begin{pspicture}(3,3)
\pnode(0,3){A}\pnode(0,0){B}\pnode(3,3){C}\pnode(3,0){D}
\transformer(A)(B)(C)(D){$\mathcal{T}$}
\end{pspicture}\begin{pspicture}(3,3)
\logic[logicType=nand,logicShowNode,logicWidth=1,
logicHeight=3,
logicNInput=6,logicChangeLR](2,1){NAND3}\end{pspicture}
```

TikZ

```
\begin{tikzpicture}  
\node(tex) at (3,2) {\TeX};  
\node(TeX-XeT) at (3,0) {\TeX-XeT};  
  
\draw(tex) to (TeX-XeT);  
\end{tikzpicture}
```

The L^AT_EX Graphics Companion

Für alles weitere:

- Herbert Voss:
- PS Tricks
- Michel Goossens, Sebastian Rahtz, Frank Mittelbach:
- The L^AT_EX Graphics Companion
- und natürlich Paketdokumentationen (extrem umfangreich!)

pgfplots

- Sehr nützliches Paket zum Plotten von Funktionen, Daten, usw. *innerhalb* des Dokumentes: `pgfplots`
- Vorteil gegenüber externen Lösungen wie gnuplot: Code ist im Dokument, alles ist portabel und man ist nicht auf externe Programme angewiesen
- Nachteil: T_EX rechnet extrem langsam ...
(vllt. bald Besserungen, wenn Rechenarbeit auf Lua übertragen werden kann)