LaTeX im Studium

PDF mit LaTeX erzeugen

Jörn Clausen
joern@TechFak.Uni-Bielefeld.DE

Übersicht

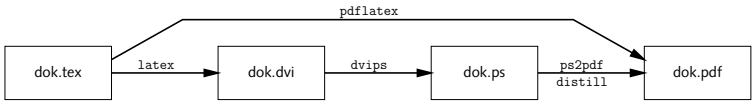
- Portable Document Format
- Wege von LaTeX zu PDF
- typische Probleme bei der Erzeugung von PDF
- Mehrwert von PDF-Dokumenten

PDF

- Portable Document Format
- Anfang der 1990er Jahre von Adobe entwickelt
- Nachfolger von PostScript
  - keine Programmiersprache mehr
  - Kompression
  - externe Bildformate (TIFF, JPEG, PNG, . . .)
  - Metadaten
  - Hypertext

PDF, cont.

- Software-Paket „Acrobat“, u.a. Reader und Distiller
- freie Alternativen:
  - Ghostscript, `ps2pdf`
  - Xpdf
- pdfTEX



Aufgaben

- Das Archiv `uebung2.tar.gz` enthält wieder einige Dateien für die heutigen Übungsaufgaben. Entpacke es an einer geeigneten Stelle.
- Erzeuge die Datei `dok.ps` mit Hilfe des Makefiles. Konvertiere die PostScript-Datei in eine PDF-Datei. Probiere den Acrobat Distiller und Ghostscript aus:

```
$ distill -pagesize 21 29.7 cm dok.ps
$ ps2pdf -sPAPERSIZE='a4 dok.ps
```

- Vergleiche die Größe der erzeugten PDF-Dateien miteinandern, und mit der Größe der Ausgangsdatei im PostScript-Format.

Aufgaben

- Der Name der Ziel-Datei kann folgendermaßen angegeben werden:

```
$ distill ... -pairs dok.ps dok.pdf
$ ps2pdf ... dok.ps dok.pdf
```

- Ergänze das Makefile um eine Suffix-Regel, um `.pdf`-Dateien aus `.ps`-Dateien zu erzeugen. Entscheide Dich für eines der beiden Konversionsprogramme.

- Suffix-Regel, um mit Hilfe von Ghostscript PDF aus PostScript zu erzeugen:

```
%.pdf: %.ps
	ps2pdf -sPAPERSIZE=a4 $< $@
```

Fonts

- Schriftart „Computer Modern“ von Donald Knuth
- viele Schnitte, viele Symbole
- 128 Zeichen pro Font (siehe `fonttable-cmr.dvi`)
- aber: keine Umlaute
- Problem: Algorithmus zur Silbentrennung
- Lösung: *enhanced coding* (EC/T1-Kodierung)
- 256 Zeichen pro Font (siehe `fonttable-ecr.dvi`)
- „European Computer Modern“ von Jörg Knappen

Aufgabe

- Übersetze die Datei `umlauts.tex` und sieh Dir das Ergebnis an. Was fällt auf? Sieh Dir den LaTeX-Code an und vergleiche ihn mit Deinen Beobachtungen.
- Binde das Paket `t1enc` ein

  ```
  \usepackage{t1enc}
  ```

  und übersetze die Datei erneut. Was passiert?
- Konvertiere die Datei nach PDF und sieh Dir das Ergebnis im Acrobat Reader an. Bewerte das Ergebnis.

Fonts, cont.

- PDF verwendet PostScript-Fonts
- zwei Arten: Type1 (Vektor-Fonts) und Type3 (Bitmap-Fonts)
- Type3-Fonts werden im Acrobat Reader sehr schlecht dargestellt
- also: Type1-Fonts verwenden
- Computer Modern: früher kommerziell, inzwischen kostenlos
- European Computer Modern: keine freie Umsetzung
- Latin Modern: erweiterte Computer Modern, 2003 veröffentlicht
- oder: andere PostScript-Fonts (Times, Palatino, Syntax, . . .)

Aufgaben

- Binde zusätzlich das Paket `lmodern` ein. Wie stellt der Acrobat Reader die resultierende PDF-Datei dar?
- Kommentiere die Pakete `t1enc` und `lmodern` aus und binde stattdessen das Paket `ae` ein. Wie sieht das Ergebnis nun aus?
- Kommentiere `t1enc` wieder ein, `ae` aus und binde das Paket `times` ein. Ändere die Zahl im `\typearea`-Befehl auf „`8`“. Wie ist die Lesbarkeit des resultierenden PDF-Dokuments?
- Vergleiche die Font-Tabellen `fonttable-ecr.dvi`, `fonttable-lmr.dvi` und `fonttable-aer.dvi` miteinander.

Eigenschaften von PDF

- Fähigkeiten von PDF ausnutzen: Hypertext, Metadaten, ...
- Komplettlösung: `hyperref` von Sebastian Rahtz
- Umdefinition vieler LaTeX-Makros (`\ref`, `\cite`, `\footnote`, ...)
- teilweise Anpassung an Zusatzpakete
- als letztes Paket einbinden

Aufgaben

- Übersetze `pdffeat.tex` und sieh Dir das Ergebnis im Acrobat Reader an. Binde das Paket `hyperref` ein und übersetze erneut. Welche Zusatzinformationen enthält das PDF-Dokument nun?
- Füge die folgenden Zeilen vor `\begin{document}` ein:

```
\hypersetup{
  colorlinks=true,
  pdfauthor={Joe User},
  pdftitle={Eigenschaften von PDF}
}
```

  Wähle im Acrobat Reader das Menu
  „File → Document Properties → Summary. . . “ aus.

- Füge in das PDF-Dokument *thumbnails* ein:

```
$ pdfthumb pdffeat.pdf pdffeat-thumbs.pdf
```

- Nur durch Einbinden von `hyperref` werden sämtliche Verweise zu Hypertext-Links. Dies betrifft das Inhaltsverzeichnis, Verweise mit `\ref` und `\cite` und Fußnoten. Die Überschriften der Abschnitte werden als *bookmarks* verwendet, so daß man schnell im Dokument navigieren kann.
- Die Links werden jetzt nicht mehr als Kästen dargestellt, sondern durch farbig markierten Text. Das Informations-Menu enthält die angegebenen Daten zu Autor und Titel, und einige weitere Informationen über den Entstehungsprozeß der PDF-Datei.

externe Links

- Verweise auf externe Dokumente wie in HTML:

```
\href{http://www.letour.fr/}{Die Tour de France}
```

- Web-Browser im Acrobat Reader konfigurieren
- spezielle Zeichen (~, ?, #, ...) können direkt verwendet werden
- schlechter Stil, auch in PDF:

  Klicken sie hier, um zur Homepage der Technischen Fakultät zu gelangen.

pdfTeX

- Modifikation von TeX, erzeugt PDF statt dvi
- Hàn Thế Thành, ehemals Universität Brno, jetzt wieder Vietnam
- kürzerer Weg zum Ziel
- unterstützt beliebige Makropakete
- aber: auf dvips angewiesene Pakete funktionieren nicht mehr
- Opfer: PSTricks, psfrag, draft
- gleiche Regeln bzgl. Type1- vs. Type3-Fonts

Aufgaben

- Übersetze die Datei `hello.tex` mit pdfLaTeX und sieh Dir das Ergebnis im Acrobat Reader an. Welche Dateien entstehen bei der Übersetzung?

- Es entstehen die gewohnten Dateien `hello.log` und `hello.aux`. Insbesondere hat die aux-Datei das gleiche Format wie bei einem „normalen" LaTeX-Lauf. Es gelten die gleichen Regeln, um z.B. Verweise korrekt aufzulösen, d.h. LaTeX bzw. pdfLaTeX muß mehrfach aufgerufen werden.

pdfTeX, cont.

- Basis-Pakete nutzen Fähigkeiten von pdfTeX aus:
  - JPEG/PNG-Grafiken mit `graphics`-Paket einbinden
  - Hypertext, Metadaten, usw. mit `hyperref`
- Ziel: Dokument mit LaTeX und pdfLaTeX übersetzen können
- einige Regeln sind einzuhalten
- bei Bedarf: Abfrage des Übersetzers

Grafiken einbinden

- gewohnte Befehle:

```
\usepackage{graphicx}

\includegraphics[width=.5\textwidth]{drawing}
```

- keine Dateiendung angeben
- Grafikformate: PNG, JPEG und PDF, nicht mehr TIFF
- EPS in „encapsulated“ PDF umwandeln:

```
$ epstopdf drawing.eps --outfile'='drawing.pdf
```

Aufgaben

- Übersetze die Datei `dok.tex` mit pdfLaTeX. Achte auf die Ausgabe beim Übersetzen. Welche Änderungen sind am Quelltext vorzunehmen?
- Wandle die Bilder `tflogo.gif` und `drawing.eps` in geeignete Formate um und binde sie zusätzlich in das LaTeX-Dokument ein. Verwende zur Konversion der Grafiken Suffix-Regeln.

PDF oder nicht PDF

- Übersetzer abfragen: Paket `ifpdf`

```
\usepackage{ifpdf}

Dieses Dokument wurde mit
\ifpdf
  pdf\LaTeX{}
\else
  \LaTeX{}
\fi
"ubersetzt.
```

Aufgaben

- Schreibe ein Makro `\Link`, das wie `\href` als Argumente einen URL und den zu verlinkenden Text enthält. Unter pdfLaTeX soll es das gleiche Ergebnis erzeugen wie `\href`. Unter LaTeX soll der URL als Fußnote gesetzt werden.

  Verwende das Paket `url` und das Makro `\url`, um den URL zu setzen.

- Makro-Definition:

```
\newcommand{\Link}[2]
  {\ifpdf\href{#1}{#2}\else#2\footnote{\url{#1}}\fi}
```

Beispiel:

```
weitere Informationen finden Sie auf der
\Link{http://www.TechFak.Uni-Bielefeld.DE/}{Homepage der
  Technischen Fakult"at}.
```