

# Installation von und Einführung in MikTeX

www.sigge.de/anleitung.pdf

Arne-Christian Sigge  
arne@sigge.de

11. Oktober 2004

## Inhaltsverzeichnis

<b>1</b>	<b>Allgemeines</b>	<b>3</b>
<b>2</b>	<b>Installation der einzelnen Komponenten</b>	<b>4</b>
2.1	Ghostview und Ghostscript	4
2.2	MikTeX	5
2.3	WinEdt	5
2.3.1	Dateiformate	6
2.3.2	Wörterbücher	6
<b>3</b>	<b>L<sup>A</sup>T<sub>E</sub>X in Action</b>	<b>7</b>
3.1	Umlaute	7
3.2	Grafiken	8
3.2.1	Das Paket float	9
3.2.2	Adobe Photoshop	10
3.2.3	Corel Draw	10
3.2.4	EXCEL und Visio	10
3.3	Grundgerüst für eine L <sup>A</sup> T <sub>E</sub> X-Datei	11
3.4	Platz sparen	11
3.4.1	ps2ps	12
3.4.2	book	12
<b>4</b>	<b>PDF-Dokumente erstellen</b>	<b>12</b>
4.1	Was ist das Problem?	12
4.2	Die Lösung: pdfLaTeX	13
4.3	Grafiken mit pdfLaTeX	15
4.4	Dokumenteinstellungen	16

<b>5</b>	<b>Known Problems and FAQs</b>	<b>17</b>
<b>6</b>	<b>Links und Literatur</b>	<b>18</b>

# 1 Allgemeines

$\LaTeX$  ist ein von Donald E. Knuth<sup>1</sup> entwickeltes Textsatzsystem zur Erzeugung (optisch) hochwertiger Dokumente. Im Gegensatz zu einem Textprogramm wie MS-Word, das nach dem WYSIWYG-Prinzip<sup>2</sup> arbeitet, wird ein  $\LaTeX$ -Dokument nicht während der Texterfassung formatiert, sondern in einem eigenen Vorgang. Das hat den Vorteil, dass man sich beim Schreiben ganz auf Inhalt und Struktur eines Textes konzentrieren kann, ohne von dem konkreten Formatierungsergebnis der aktuellen Zeile abgelenkt zu werden.  $\LaTeX$ -Dokumente können mit einem einfachen Texteditor erstellt werden. Es handelt sich um einfache Textdateien mit  $\LaTeX$ -Steuerbefehlen<sup>3</sup> die dann von einem  $\LaTeX$ -Compiler nach den Regeln der Buchdruckkunst gesetzt werden. Dabei optimiert (unter optischen Gesichtspunkten)  $\LaTeX$  immer die ganze Seite, und nicht nur eine Zeile wie es bei MS-Word der Fall ist. Dadurch sieht das Resultat deutlich schöner aus. Als Ergebnis erhält man eine geräteunabhängige (\*.dvi) Outputdatei, mit der allein man wenig anfangen kann. Sie muss zuerst in eine Postscriptdatei konvertiert werden, die dann von anderen Programmen zum Drucker geschickt oder am Bildschirm angezeigt werden kann.

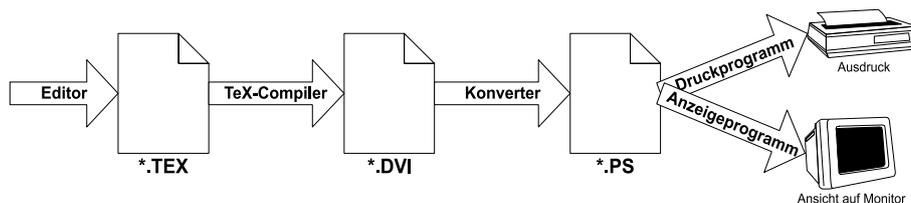


Abbildung 1: Prinzipieller Ablauf bei der Erstellung eines  $\LaTeX$ -Dokumentes

$\LaTeX$  lässt sich problemlos in einer Windowsumgebung einsetzen. Die Installation ist inzwischen kinderleicht, wenn man weiß, was man denn alles braucht. Die notwendigen Programme sind zwar (fast) alle frei im Internet verfügbar, stammen aber nicht aus einer Hand und müssen daher zusammengesucht werden. Die Installation, Konfiguration und der Einsatz einer Kombination aus drei Produkten (MikTeX, Ghostview und WinEdt), die sich bewährt hat, wird im folgenden vorgestellt.

<sup>1</sup>Und von Leslie Lamport erweitertes

<sup>2</sup>*What You See Is What You Get*

<sup>3</sup>Dieses Dokument dient nicht dazu, die  $\LaTeX$ -Syntax zu erläutern. Dazu gibt es beispielsweise das Dokument `12kurz.pdf`, das im Internet auf den Seiten von [www.dante.de](http://www.dante.de) zu finden ist, oder die 7-Minuten Dokumente von Peter Wolf (siehe Kapitel 6).

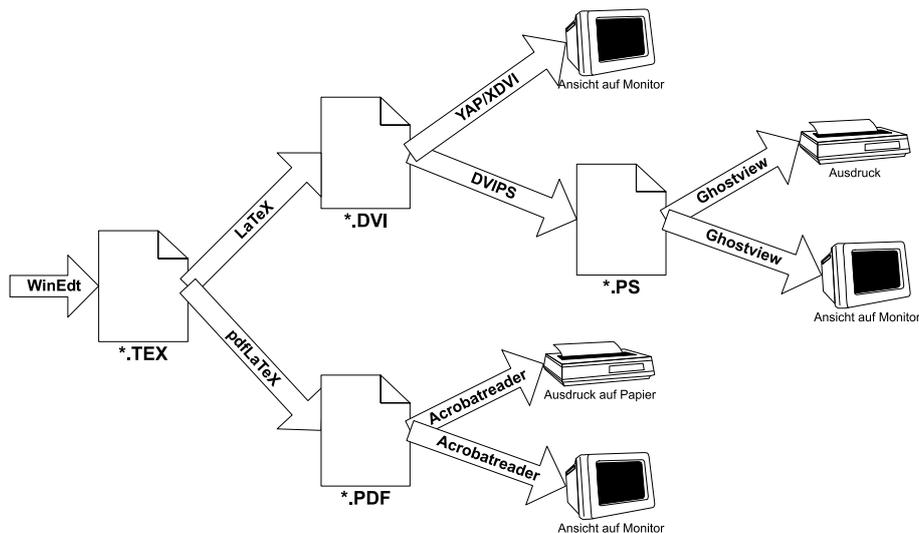


Abbildung 2: Überblick über das (mögliche) Zusammenwirken der einzelnen Programmkomponenten

## 2 Installation der einzelnen Komponenten

### 2.1 Ghostview und Ghostscript

Die Programme *Ghostview* und *Ghostscript* treten im Doppelpack auf und ergänzen sich. *Ghostscript* ist ein Interpreter für Postscriptdateien. *Ghostview* ist ein Preview und Printprogramm, das die Dienste von Ghostscript nutzt, um Postscriptdateien auf dem Bildschirm anzuzeigen oder um sie an einen nicht postscriptfähigen Drucker zu schicken. Das Programm ist ein Freewareprogramm und kann im Internet unter:

- [www.cs.wisc.edu/~ghost/](http://www.cs.wisc.edu/~ghost/)

heruntergeladen werden. Zur Installation werden zwei Dateien von dieser Homepage benötigt: `gsv46w32.exe`<sup>4</sup> und `gs814w32.exe`<sup>5</sup>. Zuerst sollte *Ghostscript*, anschließend *Ghostview* in die von den Setupprogrammen vorgeschlagenen Verzeichnisse installiert werden. Dieses Programmdoppel sollte vor der eigentlichen Installation von MiKTeX installiert werden, da MiKTeX die von *Ghostscript* installierten Zeichensätze einbindet.

<sup>4</sup>Für die im Mai 2004 aktuelle Version 4.6 von Ghostview. Andere Versionen sind entsprechend bezeichnet.

<sup>5</sup>Bezeichnet ebenfalls die im Juli aktuelle Version 8.14 von Ghostscript.

## 2.2 MikTeX

MikTeX liegt derzeit in der Version 2.4 vor. Es handelt sich ebenfalls um ein Freewareprogramm. Einen Link zu einem Downloadverzeichnis findet sich entweder direkt auf der offiziellen Homepage oder auf jedem guten FTP-Server in dem unten angegebenen Verzeichnis:

- [www.miktex.org](http://www.miktex.org)
- [ftp.uni-bielefeld.de/pub/tex/systems/win32/miktex/](ftp://uni-bielefeld.de/pub/tex/systems/win32/miktex/)

Je nachdem ob MikTeX in einem einzigen großen ZIP-File ausgeliefert wird oder in einer Sammlung von mehreren Dateien, muss die Installationsdatei noch entpackt werden. Vorsicht: Werden mehrere Dateien, darunter viele ZIP-Dateien ausgeliefert, dürfen diese **nicht** entpackt werden. Das übernimmt später das Installationsprogramm. Nur wenn eine einzelne, über 100 MB große Datei geliefert wird, ist ein manuelles entpacken notwendig. Die Installation wird durch Starten des Programms `Setup.exe`<sup>6</sup> angestoßen. Auch hier kann man den Anweisungen am Bildschirm getrost folgen und die vorgeschlagenen Verzeichnisse und Einstellungen übernehmen. Ab Version 2.1 bietet MikTeX, ähnlich wie der Internet-Explorer, eine Installation mit direktem Download der benötigten Dateien aus dem Internet an. Es kann aber auch ganz normal aus einem Verzeichnis installiert werden. Zum Abschluss der Installation muss der Rechner einmal neu gestartet werden.<sup>7</sup> Die in Abbildung 2 angesprochen Hilfsprogramme `dvips` und `YAP` sind in der MikTeX-Distribution enthalten und zu diesem Zeitpunkt bereits installiert.

Im Verwaltungsprogramm von MikTeX ist eine Komponente enthalten, die automatisch nach Updates im Internet suchen kann.

Wer Probleme bei der Installation hat, findet unter [www.miktex.org/docs.html](http://www.miktex.org/docs.html) eine etwas ausführlichere Installationsanleitung.

## 2.3 WinEdt

Um etwas komfortabler mit MikTeX zu arbeiten, empfiehlt es sich, einen speziell auf  $\text{\LaTeX}$  zugeschnittenen Editor zu verwenden. Natürlich tut es sonst auch jeder herkömmliche Texteditor, der Texte im MS-DOS Format verarbeiten kann. WinEdt ist ein Shareware-Programm und kann unter:

- [www.winedt.com](http://www.winedt.com)

bezogen werden. WinEdt ist für die Zusammenarbeit mit MikTeX schon optimiert. Bei der Installation wird die installierte MikTeX-Version erkannt und

---

<sup>6</sup>in älteren Versionen auch `setupwiz.exe`

<sup>7</sup>Ab Windows 2000 nicht mehr unbedingt notwendig, kann aber nie schaden

WinEdt so konfiguriert, dass man (fast) sofort mit der Arbeit am ersten  $\text{\LaTeX}$ -Dokument loslegen kann. Einige Einstellung sind trotzdem noch vorzunehmen. Nachdem diese Einstellungen abgeschlossen sind, sollte man nicht vergessen, diese unter dem Menüpunkt **Options - Save Settings** zu sichern.

### 2.3.1 Dateiformate

Da einfache Textdateien (und nichts anders sind ja die  $\text{\LaTeX}$ -Dokumente) unter MS-DOS und Windows von Microsoft mit unterschiedlichen Zeichensatztabellen bedacht wurden, müssen wir in das Handling dieser Dateien eingreifen. WinEdt ist eine Windows-Anwendung, MikTeX läuft (wenn auch unbemerkt) als MS-DOS-Anwendung. Unter dem Menüpunkt **Options - Settings - Translation** ist für **READ** durch Anlicken die Option **OEM => ANSI** auszuwählen. Für **WRITE** entsprechend **ANSI => OEM**.

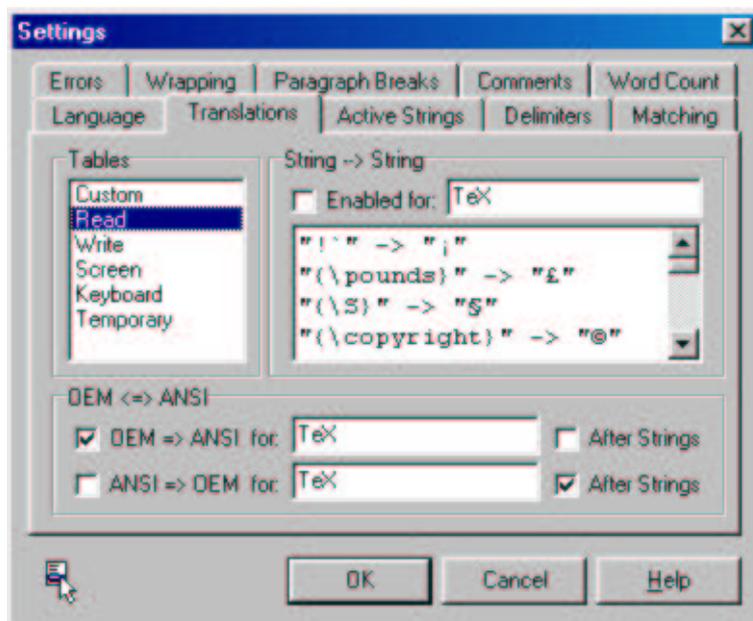


Abbildung 3: Einstellungen zur Konvertierung der Zeichentabellen beim Lesen und Schreiben der Dateien

Diese Einstellungen stellen sicher, dass Umlaute direkt im Editor eingegeben werden können und auch so in der `*.tex` Datei abgespeichert werden, dass sie unter MS-DOS lesbar sind.

### 2.3.2 Wörterbücher

Da WinEdt auch eine brauchbare Rechtschreibkorrektur anbietet, sollte man zusätzlich ein deutsches Wörterbuch installieren. Unter **Options - Dictionary**

kann man eine Wörterbuchdatei, die man zuvor in das entsprechende Verzeichnis kopiert hat<sup>8</sup>, einbinden. Dazu fügt man in die Dictionary Liste mit einem Click auf die rechte Maustaste und der Wahl des Befehls **INSERT** ein neues Wörterbuch hinzu. Nachdem man einen Namen<sup>9</sup> vergeben hat, aktiviert man es mit der Option **enabled** und sorgt mit einem Haken vor **Load on Start** dafür, dass die Inhalte beim Programmstart geladen werden.

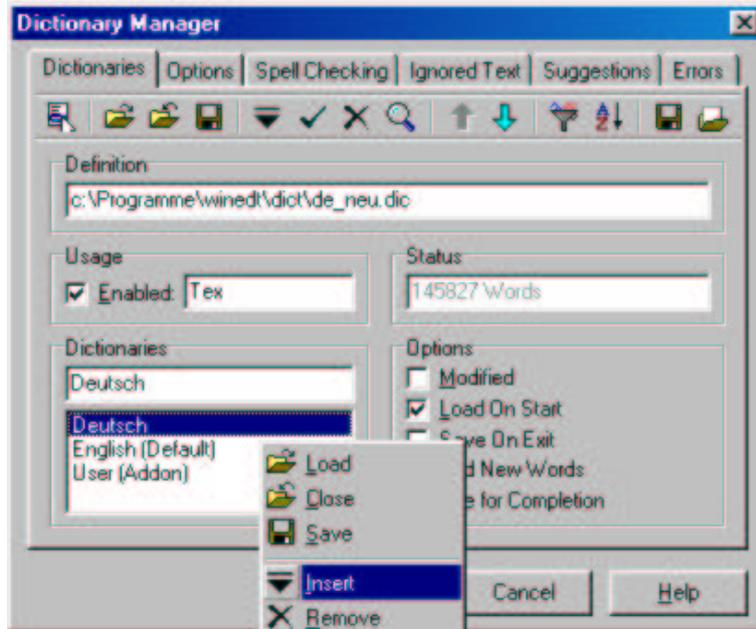


Abbildung 4: Hinzufügen der deutschen Wörterbuchdatei

## 3 $\LaTeX$ in Action

### 3.1 Umlaute

Umlaute in  $\LaTeX$ -Dokumenten bereiteten früher häufig Probleme. Etwas verwirrend sind die zahlreichen Möglichkeiten,  $\LaTeX$  dazu zu bekommen, mit Umlauten zu arbeiten. Am einfachsten (und auch satztechnisch am schönsten) ist es,  $\LaTeX$  mitzuteilen, in welcher Codierung die Quelldatei vorliegt und welche Zeichensatztabellen verwendet werden sollen. In der Präambel eines Dokumentes sollte daher definiert werden:

```
\usepackage[cp850]{inputenc}
```

<sup>8</sup>Eine entsprechende Wörterbuchdatei kann man in einem Zip-File von der WinEdt Homepage herunterladen. Die benötigte Datei für die neue deutsche Rechtschreibung heißt `de-neu.dic`

<sup>9</sup>Sinnigerweise etwas wie *Deutsch*

Hier wird die Inputdatei als eine MS-DOS Textdatei (Codepage 850)<sup>10</sup> festgelegt. Ob eine solche Datei tatsächlich vorliegt, kann man ganz leicht feststellen: Man öffnet die Datei unter MS-DOS mit dem Befehl `EDIT` und sollte Umlaute sehen. Achtung: Unter Windows werden Textdateien in einem anderen Format gespeichert. Umlaute, die mit dem Windows `notepad` Editor zu sehen sind, verursachen später einen Fehler.

Die T1-Fonts, die auch echte Umlaute<sup>11</sup> enthalten, werden aktiviert mit:

```
\usepackage[T1]{fontenc}
```

Um die deutsche Silbentrennung und einige andere Besonderheiten der deutschen Sprache zu aktivieren und das A4-Papierformat auszunutzen, ist es noch sinnvoll

```
\usepackage{ngerman, a4}
```

zu verwenden.

## 3.2 Grafiken

Das Einbinden von Grafiken stellt viele TeX-Einsteiger vor schier unlösbare Probleme, weil man eben nicht einfach eine JPEG-Datei per Drag'n'Drop in sein Dokument integrieren kann. L<sup>A</sup>T<sub>E</sub>X benötigt EPS-Dateien, eine Sonderform der bekannten Postscriptdateien. EPS-Dateien lassen sich leicht aus den meisten Programmen heraus erzeugen. Allerdings sind einige Dinge zu beachten. Postscript ist ein Standard, den jeder ein klein wenig anders sieht und um eigene, proprietäre Merkmale ergänzt. L<sup>A</sup>T<sub>E</sub>X versteht nur ganz reine EPS-Dateien ohne zusätzlichen Schnickschnack. Die Daten sollten ASCII codiert abgespeichert werden, etwaige Bildvorschauen oder sonstige Metadaten dürfen nicht integriert sein. Die meisten Programme bieten diese Einstellmöglichkeiten beim Export von (E)PS-Dateien. Einige Hinweise dazu finden sich in Abschnitt 3.2.2 ff.

Ob die EPS Datei richtig erzeugt wurde, lässt sich mit einem einfachen Texteditor feststellen. Einfach die EPS Datei mit dem Editor öffnen und die ersten Zeilen betrachten. So sieht der Anfang einer korrekten EPS-Datei aus:

```
%!PS-Adobe-3.0 EPSF-3.0
%%BoundingBox: 63 687 171 766
%%LanguageLevel: 1
%%Creator: CorelDRAW 10
%%Title: Grafik.eps
```

---

<sup>10</sup>Man könnte auch eine ANSI-Datei angeben mit dem Parameter `ansinew`, aber ich halte mich in meiner Beschreibung an MS-DOS, da dies die wenigsten Seiteneffekte verursacht.

<sup>11</sup>Mit `\u{a}` wird nur ein `a` mit zwei Punkten zu einem Umlaut zusammengebastelt

```

%%CreationDate: Thu Oct 31 11:02:36 2002
%%For: Arne Sigge
%%DocumentProcessColors: Black
%%DocumentSuppliedResources: (atend)
%%EndComments
%%BeginProlog
/AutoFlatness false def /AutoSteps 0 def ...

```

Wurde die Datei (für die Anforderungen von L<sup>A</sup>T<sub>E</sub>X) falsch erzeugt, findet sich zu Beginn der Datei binärer „Zeichenmüll“ bestehend aus ausländischen Zeichensymbolen und kleinen Quadraten oder anderen geometrischen Figuren. Ist eine korrekte EPS-Datei erst einmal vorhanden, lässt sie sich relativ problemlos in ein L<sup>A</sup>T<sub>E</sub>X-Dokument einbinden. Dazu benötigt man das Package `graphicx`. Die Befehle zum Einbinden der Grafik lauten dann:

```

\usepackage{graphicx}
...
\begin{document}
...
\includegraphics[width=8cm]{dateiname.eps}

```

Der Parameter `width=8cm` ist optional. Es können weitere Parameter durch Komma getrennt angegeben werden. Eine genaue Beschreibung der möglichen Parameter findet sich in dem Dokument `grfguide.pdf`, das jeder MiKTeX-Installation beiliegt.

Da es sich bei Postscript um eine Seitenbeschreibungssprache handelt, werden grafische Abbildungen (also vektororientierte wie beispielsweise ein Histogramm) wesentlich effizienter abgespeichert als pixelorientierte (beispielsweise ein Screenshot oder jede JPEG-Datei). Bei der Erzeugung einer EPS-Datei sollte dies bedacht werden. Der Umweg über eine JPEG- oder GIF-Datei führt nicht nur zu qualitativen Nachteilen, sondern auch zu erheblich größeren Dateien.

### 3.2.1 Das Paket `float`

Da Bilder in L<sup>A</sup>T<sub>E</sub>X-Dokumenten als Flussobjekte angesehen werden, kann es passieren, dass die Bilder nicht unbedingt an der Stelle landen, an der sie vom Autor vorgesehen waren. Um den Fluss der Bilder zu steuern gibt es das Paket `float`. Mit ihm ist es möglich, festzulegen, dass ein Bild an genau der vorgesehenen Stelle im Text verbleibt. Dazu wird einer `figure`-Umgebung der Parameter `[H]` mitgegeben. In einer `figure`-Umgebung lassen sich auch Bildzeilen erstellen. Der Befehl `\centerline` im folgenden Beispiel dient der Zentrierung des Bildes auf der Seite:

```

\begin{figure}[H]
\centerline{\includegraphics[width=8cm]{bild.eps}}
\caption{Hier steht der Text der Bildzeile}
\end{figure}

```

### 3.2.2 Adobe Photoshop

Photoshop eignet sich sehr gut um JPEG-, GIF-Dateien oder direkt einen Screenshot in eine EPS-Datei umzuwandeln. Im Menü **Speichern unter** findet sich der Eintrag **Photoshop-EPS**. Das Exportmenü erlaubt es, unnötige und hinderliche Features abzustellen:

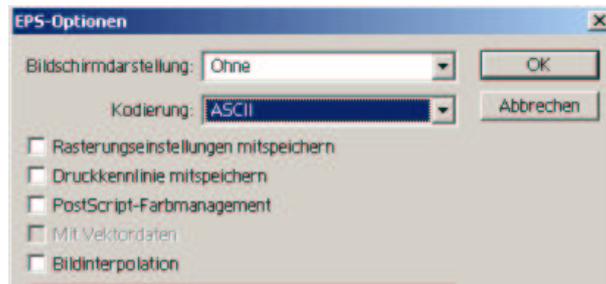


Abbildung 5: Einstellungen für den EPS-Export unter Photoshop

Eine weitere Möglichkeit EPS oder auch PDF Dateien aus JPEG Dateien zu erstellen bietet die Freeware-Toolsammlung Imagemagick. Mit ihr lassen sich Bilddateien in unterschiedlichen Formate konvertieren.

### 3.2.3 Corel Draw

Das Grafikprogramm Corel Draw bietet ebenfalls sehr einfache Möglichkeiten, eine EPS-Datei zu erstellen. Die gewünschten Objekte werden markiert und dann exportiert. Als Exportformat ist **Encapsulated Postscript (EPS)** zu wählen. Auch hier sind wieder alle Optionen (Vorschaugrafik, einbetten von Schriftarten etc.) abzuwählen.

### 3.2.4 EXCEL und Visio

Der einfachste Weg, eine EXCEL- oder Visio-Grafik in eine EPS-Datei zu konvertieren, ist der folgende:

1. Grafik in EXCEL oder Visio erstellen und markieren
2. mit **STRG-C** in die Zwischenablage kopieren
3. in das Programm Corel Draw wechseln

4. Grafik über **Inhalte einfügen**<sup>12</sup> aus der Zwischenablage importieren
5. Grafik ggf. etwas nachbearbeiten
6. exportieren wie in 3.2.3 beschrieben.

Wem dieser Weg nicht geheuer ist oder wem Corel Draw fehlt, kann auch eine PS-Druckertreiber installieren und versuchen, über den Druck in eine Datei eine EPS-Grafik zu erstellen. In diesem Fall ist aber einiges Herumprobieren notwendig, um zu den gewünschten Ergebnissen zu kommen<sup>13</sup>. Der in Visio vorgesehene Export in EPS-Dateien erstellt keine standardkonformen EPS-Dateien.

Tabellen lassen sich aus EXCEL heraus mit dem Freeware-Tool XL2LaTeX von direkt als L<sup>A</sup>T<sub>E</sub>X-Datei abspeichern. xl2LaTeX ist auch auf dem CTAN-Server zu finde. Es handelt sich um ein Plugin für EXCEL.

### 3.3 Grundgerüst für eine L<sup>A</sup>T<sub>E</sub>X-Datei

Fasst man die Hinweise aus den oberen Abschnitten zusammen, ergibt sich folgendes Grundgerüst für eine einfach L<sup>A</sup>T<sub>E</sub>X-Datei, die deutsche Umlaute, das A4-Format und das Einbinden von Bilder unterstützt.

```
\documentclass[12pt]{article}
\usepackage[cp850]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{ngerman, a4}
\usepackage{float}
\usepackage{graphicx}
\begin{document}

\end{document}
```

### 3.4 Platz sparen

Manchmal ist es ganz nützlich, Zwischenversionen seines Dokumentes auszudrucken. Wer dabei Papier sparen will, kann mit relativ geringem Aufwand zwei Batchdateien anlegen, die eine Postscriptdatei so konvertieren, dass im die Seiten mit dem Faktor 0.71 (A4 auf A5) skaliert, gedreht und paarweise auf eine A4 Seite gestellt werden. Die notwendigen Tools sind in jeder MikTeX-Umgebung enthalten und müssen nur entsprechend angesteuert werden.

---

<sup>12</sup>Nicht einfach über STRG-V!

<sup>13</sup>Support ist für diese Lösung vom Autor nicht zu erwarten

### 3.4.1 ps2ps

In der einfachen Variante werden die Seiten von A4c auf A5 verkleinert, um 90 Grad gedreht und paarweise auf einer A4 Seite nebeneinander gestellt. Dazu legt man eine Batchdatei namens `ps2ps.bat` im Verzeichnis `...miktex\bin\` an. In ihr sollte folgende Zeile stehen:

```
pstops 2:0L@0.7(21cm,0)+1L@0.7(21cm,14.85cm) %1 %2
```

Aufgerufen wird die Datei dann mit `ps2ps.bat Inputdatei.ps Outputdatei.ps`

### 3.4.2 book

In der etwas komplizierteren Variante entstehen zwei Postscriptdateien. Die erste Datei (`1.ps`) enthält die Vorderseiten, die zweite Datei (`2.ps`) die Rückseiten eines Dokumentes im Buchdruck. Schafft man es, die Papierführung seines Druckers zu durchschauen, kann man so 4 A5-Seiten (zwei vorne, zwei hinten) auf ein A4 papier drucken und so in der Mitte heften, dass ein handliches Büchlein entsteht. Eine Batchdatei mit dem Namen `book.bat`, sollte diese Zeilen enthalten:

```
psbook %1 b.ps  
psnup -2 b.ps h.ps  
psselect -o h.ps 2.ps  
psselect -e h.ps 1.ps
```

Natürlich muss auch sie an einem Ort stehen, an dem das System sie finden kann, beispielsweise im `...bin`-Verzeichnis von MikTeX. Für die ersten Versuche sollte man eine kleinere Datei (natürlich minimal acht Seiten) wählen. Beim Aufruf muss nur der Name der Inputdatei als Parameter übergeben werden. Die beiden Outputdateien haben feste Namen.

## 4 PDF-Dokumente erstellen

### 4.1 Was ist das Problem?

Das PDF Format hat sich in den letzter Zeit als Dokumentformat im Internet durchgesetzt. Es stammt ebenfalls aus dem Hause Adobe, das auch schon die Entwicklung von Postscript maßgeblich beeinflusst hat. PS- und PDF-Dokumete sind sich prinzipiell sehr ähnlich. Bei beiden handelt es sich um einfache Textdateien, die (Text-)Dokumente, in einer eigenen Sprache abbilden, die dann ein Zielgerät (meist ein Drucker) interpretieren kann. Durch Einbindung von Zeichensätzen ist es möglich, Dokumente ohne Veränderungen der Optik (ein wesentlicher Unterschied zu SGML und HTML!) auf elektronischem Wege weiterzugeben. PS-

und PDF-Dokumente enthalten Informationen über den Text, die verwendeten Zeichensätze und die Aufteilung, des Textes auf die unterschiedlichen Seiten. Sie enthalten kein fertiges (pixelorientiertes) Bild der Seite. So kann jedes Ausgabegerät entsprechend seinen Fähigkeiten die Seite aufbauen. Während PS-Dokumente hauptsächlich genutzt werden, um Dokumente für den Druck (und damit ist nicht nur der Ausdruck auf dem heimischen Drucker, sondern auch der Buchdruck in einer professionellen Druckerei gemeint) bereitzustellen, sind PDF-Dokumente für den Vertrieb über das Internet optimiert. Sie benötigen weniger Speicherplatz und können interaktive und multimediale Elemente enthalten wie Links zu Webseiten, eine interne Navigationslogik oder auch Töne. Ihr Nachteil besteht in einer wesentlich geringeren Druckqualität als sie PS-Dokumente besitzen. Durch die große Verbreitung des kostenlosen Acrobat Readers können sie aber von fast jedem Computerbesitzer problemlos am Bildschirm gelesen und ausgedruckt werden. Viele (verbreitungswürdige) Dokumente entstehen als  $\text{\LaTeX}$ -Dokument. Eine PS-Datei aus einem  $\text{\LaTeX}$ -Dokument zu erzeugen ist kein Problem, daher sollte es auch ein Leichtes sein, PDF-Dokumente aus in  $\text{\LaTeX}$  geschriebenen Dokumenten zu generieren. Viele Wege führen zum Ziel. Es gibt prinzipiell zwei Wege um PDF-Dokumente aus  $\text{\LaTeX}$ -Quelldokumenten zu erzeugen. Eine Möglichkeit besteht in der Konvertierung einer herkömmlich erzeugten PS-Datei in ein PDF-Dokument. Der Vorteil liegt darin, dass man sich in keinster Weise umstellen braucht und lediglich die vorhandenen PS-Dateien konvertiert. Diese Methode beinhaltet allerdings einige Nachteile:

- Zur Konvertierung ist ein kostenpflichtige Version des Adobe Acrobat Distillers erforderlich.
- Es sind mehrere Schritte durchzuführen um das PDF-Dokument zu bekommen ( $\text{\LaTeX}$ , DVIPS, Distiller).
- Zahlreiche PDF-Features können nicht genutzt werden.
- DVIPS bindet Zeichensätze pixelorientiert in die PS-Dateien ein, die für den Zieldrucker optimiert sind. Das führt dazu, dass die Schrift im PDF-Dokument als Abfolge von groben Bildern der einzelnen Buchstaben erscheint, somit unschön aussieht und das Dokument nicht durchsuchbar ist. Das Resultat sieht so aus:

**Dies ist ein PDF-Test mit DVIPS und Distiller, öäüÄÖÜß**

## 4.2 Die Lösung: pdfLaTeX

Die andere Möglichkeit ist die Verwendung von **pdfTeX** bzw. **pdfLaTeX**. Dieses Tool ist in zahlreichen  $\text{\LaTeX}$ -Distributionen enthalten und arbeitet als eigener

TeX-Compiler. Es erzeugt aus \*.tex Dateien direkt PDF-Dokumente. Dieser Weg hat Vorteile:

- pdfTeX ist kostenlos erhältlich
- Der Umweg über DVI- und PS-Dateien entfällt
- Spezielle PDF-Features sind realisierbar
- pdfTeX bindet PS Type 1 Zeichensätze mit in die PDF-Dokumente ein. Das Resultat sieht deutlich besser aus:

Dies ist ein PDF-Test mit pdfTeX. öäüÄÖÜß

...aber auch einige Nachteile:

- Eine Ergänzung der eigenen TeX-Installation ist ggf. notwendig.
- Vorhandene L<sup>A</sup>T<sub>E</sub>X-Dokumente müssen ggf. geringfügig angepasst werden.
- Exotische Packages und Zeichensätze werden eventuell nicht unterstützt.

Wer von den Vorteilen überzeugt und von den Nachteilen nicht abgeschreckt ist, kann in den folgenden Absätzen erfahren, welche Schritte durchzuführen sind um pdf(La)TeX zu verwenden.

Zusätzlich zu der oben dargestellten MikTeX Umgebung werden benötigt:

- AE-Fonts und Style
- Postscript Versionen der CM-Fonts (z.B. von Bakoma)
- `epstopdf` zum Umwandeln von EPS- in PDF-Dateien

All diese Komponenten finden sich auf praktisch jedem CTAN-Server<sup>14</sup>. Seit der MikTeX-Version 2.1 sind der AE-Style und die PS-Fonts im Lieferumfang enthalten und werden bei der Installationsvariante *large* auch gleich installiert. Um die Fonts und den AE-Style nachträglich zu installieren, müssen die beiden ZIP-Dateien mit ihrer Verzeichnisstruktur entpackt werden und an bestimmte Stellen in den Ordner `localtexmf` kopiert werden. Achtung: Teilweise müssen noch Unterordner angelegt werden wie z.B. `fonts`.

Der Ordner `ae` wird in den Unterordner `\localtexmf\tex\latex` kopiert. Die Ordner `tfm` und `vf` kommen in den Unterordner `\localtexmf\fonts`. Die Ordner `afm`, `tfm`, `pfb` und `pfm` aus den Ordnern `Bakoma` und `Bluesky` werden ebenfalls

---

<sup>14</sup>[www.ctan.org](http://www.ctan.org)

in den Unterordner `\localtexmf\fonts` kopiert. Anschließend sollte man über das MikTeX-Menü die Filename Database refreshen.

Um pdfLaTeX nutzen zu können muss, bei der Wahl von `\documentclass` der Parameter `[pdftex]` übergeben werden. Wichtig ist der Befehl `\usepackage{ae}`, der die Verwendung von AE-Fonts steuert, die benötigt werden um die T1 Fonts in PDF-Dokumenten realisieren zu können. Damit ist eigentlich auch schon alles gesagt, was es zu beachten gibt bei der Umstellung der alten LaTeX Dokumente. Eine einfache Quelldatei für pdfTeX sieht demnach so aus:

```
\documentclass[a4paper, pdftex]{article}
\usepackage{german}
\usepackage[cp850]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{ae}
\begin{document}
Dies ist ein PDF-Dokumenttest äöüÄÖÜß
\end{document}
```

Ein so aufgebautes Dokument kann problemlos von  $\LaTeX$  und pdf(La)TeX verarbeitet werden. Diese Datei enthält noch keine PDF-Specials, auf die ich hier auch nicht weiter eingehen möchte. Sie werden im beiliegende Handbuch `pdftexman.pdf` ausführlich beschrieben.

### 4.3 Grafiken mit pdfLaTeX

Grafiken in LaTeX waren ja schon immer ein Problem, wenn sie nicht als PS oder EPS Dateien vorlagen. pdfTeX macht hier einiges einfacher, verkompliziert die Welt aber auch etwas. pdfTeX kann Bildformate wie JPG, PNG und PDF direkt einbinden, hat jedoch Probleme mit EPS-Dateien. Das macht die Sache etwas komplizierter, aber nicht unlösbar. Benutzer des `graphicx` Packages brauchen sich praktisch nicht umstellen. pdfTeX arbeitet mit `graphicx` problemlos zusammen. Bilder können mit dem Befehl `\includegraphics[width=10cm]{datei.jpg}` eingebunden werden. Wer auf diese Weise auch EPS oder PS Grafiken eingebunden hat, braucht sich nur ein wenig umzustellen. Da pdfTeX nicht mit EPS-Grafiken, dafür aber mit PDF Grafiken umgehen kann, konvertiert man die EPS-Grafiken einfach in PDF-Grafiken mit dem Tool `epstopdf`. Möchte man Dokumente erstellen, die sowohl mit LaTeX als auch mit pdfTeX compilierbar sein sollen, muss man an dieser Stelle aufpassen! Da  $\LaTeX$  und pdfTeX nur unterschiedliche Grafikformate verarbeiten können, sollte man bei der Angabe der Grafikdatei im Befehl `\includegraphics` auf die Dateierweiterung verzichten und im entsprechenden Verzeichnis beide Dateiformate mit gleichem Namen aber unterschiedlichen Extensionen (also `dateiname.eps` und `dateiname.pdf`) liegen haben. Das ist zwar ein etwas unschöner, aber wirkungsvoller Weg. Nun braucht man das Quelldokument nur an einer leicht zu findenden Stelle für den gewünschten Compiler

anzupassen. Der Parameter [pdftex] vor dem Befehl \documentclass ist für den Durchlauf mit pdfTeX zu setzen, vor der Compilierung mit L<sup>A</sup>T<sub>E</sub>X jedoch wieder zu entfernen. Eine Alternative sind die PFD-Specials zum Einbinden von Grafiken. Verwendet man sie, kann man die Datei allerdings nur noch mit pdfTeX compilieren! Um eine Grafik einzubinden, sind drei Befehle notwendig. Der erste stellt eine Verknüpfung mit der Bilddatei her, der zweite bindet die Datei über eine Referenznummer ein, die vom dritten Befehl geliefert wird. Klingt etwas kompliziert, ist aber ganz einfach:

```
\pdfximage{Test.JPG}\pdfrefximage\pdflastximage
```

## 4.4 Dokumenteinstellungen

Zum Abschluss noch eine Dokumentvorlage für das problemlose Nebeneinander von L<sup>A</sup>T<sub>E</sub>X und pdfLaTeX. Fügt man diese Codezeilen am Anfang des Dokumentes ein, erkennt das Dokument, ob es gerade von L<sup>A</sup>T<sub>E</sub>X oder pdfLaTeX bearbeitet wird und verhält sich entsprechend. Umstellungen sind nicht mehr notwendig.

```
\newif\ifpdf
\ifx\pdfoutput\undefined
\pdffalse % Wir benutzen PDFLaTeX nicht
\else
\pdfoutput=1 % Wir benutzen PDFLaTeX
\pdftrue
\fi
\documentclass[12pt,a4paper]{article}
\usepackage[cp850]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{german, a4}
\ifpdf
  \usepackage{ae}
  \usepackage[pdftex]{graphicx}
  \usepackage[pdftex,
    pagebackref,
    colorlinks=true,
    citecolor=magenta,
    pdftitle={Title},
    pdfauthor={Author},
    pdfsubject={Subjekt},
    pdfkeywords={Keyword},
    bookmarksopen=false
    ]{hyperref}
\pdfcompresslevel=1
```

```

\else
  \usepackage{graphicx}
\fi
\title{MikTeX 2.0 Installationsanleitung}
\author{Arne-Christian Sigge}
\date{\today}
\begin{document}
...
\end{Document}

```

## 5 Known Problems and FAQs

Was tun, wenn etwas nicht so klappt wie gewünscht? Auch hier gibt es eine einfache Vorgehensweise:

1. Gründlich über das Problem nachdenken
2. Diesen Abschnitt lesen
3. Nochmals gründlich nachdenken
4. Dieses Dokument nochmals vollständig lesen
5. Auf der MikTeX-Homepage oder in einer Newsgroup nach einer Lösung suchen (z.B. über [groups.google.com](http://groups.google.com))
6. Gegebenenfalls auch Dokumente anderer Autoren lesen
7. Jemanden mit Erfahrung fragen

Wichtig ist es, diese Reihenfolge genau einzuhalten. Ein ausführliche Liste mit FAQs findet sich natürlich auch auf der MikTeX-Homepage. Einige Probleme, die in letzter Zeit verstärkt aufgefallen sind:

**Einige \*.dll werden bei der Installation nicht gefunden:** Die Installation will nicht starten, da einige DLLs, die zum Betriebssystem gehören, nicht gefunden werden. Das passiert gerne bei älteren Betriebssystemen wie Windows 95, 98 oder NT, sollte jedoch nicht passieren bei Windows ME, 2000 und XP. Das Einspielen der aktuellen Servicepacks sollte diese Probleme beseitigen. Andernfalls finden sich auf der MikTeX-Homepage Links (unter Installation-Instruktionen) zu Downloadmöglichkeiten der entsprechenden Dateien auf der MS-Homepage.

**MikTeX ist nicht da:** MikTeX will nicht starten, da es nicht gefunden wird (unter WinEdt geht nur kurz eine Fenster auf). Ist der Rechner auch nach der Installation einmal neu gestartet worden? Wenn man an der MS-DOS Eingabeaufforderung den Befehl `latex` eingibt, sollte sich MikTeX melden (Abbruch dann mit `STRG-C`). Falls nicht, sollte in der `autoexec.bat` überprüft werden, ob der Pfad auf das MikTeX-Verzeichnis auch gesetzt ist (z.B.: `set PATH=c:\miktex\bin;%PATH%`).

**MikTeX findet etwas nicht:** Meist werden nachträglich installierte Zeichensätze oder `sty`-Files nicht gefunden. In vielen Fällen hilft ein `Refresh` `Filename Database` aus dem MikTeX-Options Programm.

**YAP findet Ghostscript nicht:** YAP benötigt mindestens Ghostscript 6.0 um die eingebetteten Grafiken anzeigen zu können. Wenn man also unbedingt Grafiken in der Vorschau sehen möchte, installiert man eine aktuelle Version von Ghostscript.

**epstopdf findet Ghostscript nicht:** Die Datei `gswin32.exe` muss ebenfalls in einem Suchpfad der `PATH`-Variablen zu finden sein.

**Mein Bild wird nicht angezeigt:** Ist die EPS-Datei korrekt erstellt? Öffne die EPS-Datei mit einem Texteditor und suche nach binärem Zeichenmüll.

**Wie mache ich Anführungszeichen?** Anführungszeichen werden mit `"`, Abführungszeichen mit `'` realisiert.

**Wie mache ich einen Absatz?** Ein Absatz wird mit einer Leerzeile erzeugt. **Keinesfalls** sollte man den Befehl `Zeilenende` `\\` (wohlmöglich noch zweifach) verwenden! Ein vorzeitiges `Zeilenende` ist kein Absatz!

**Wie binde ich Quellcode ein?** Hier helfen Packages wie `Verbatimfiles` oder `Listings`

**Wie stelle ich URLs da?** Entweder in einer `\texttt{}` Umgebung (funktioniert auch in Fußnoten). Dann müssen besondere Zeichen ggf. erzeugt werden. Beispiel: Tilde (`~`) durch `\char'176` oder ein Unterstrich durch `\_`. Ein Zeilenumbruch bei langen URLs wird durch ein Leerzeichen an passender Stelle erreicht. Alternativ kann man eine `\verb` Umgebung verwenden. Beispiel: `\verb+http://www...+`. Ein Zeilenumbruch lässt sich durch eine Aufspaltung in mehrere `Verb`-Umgebungen erzielen.

## 6 Links und Literatur

Zum Abschluss noch eine sicherlich unvollständige, aber doch schon sehr hilfreiche Liste mit Links zu Downloadmöglichkeiten und Hilfestellungen im Internet, sowie

einige Literaturhinweisen. Wer eine kurze Einführung in  $\LaTeX$  benötigt, sollte auf der Homepage von Peter Wolf die beiden 7-Minuten Dokumente zum Thema lesen, anschließend auf der **Dante**-Seite nach dem Dokument `l2kurz.pdf` suchen. Wer sich ein Buch zulegen möchte, macht mit dem *Kopka* als Standardwerk sicher nichts falsch.

- [www.cs.wisc.edu/~ghost/](http://www.cs.wisc.edu/~ghost/): Homepage des *Ghostscript*-Projektes. Hier finden sich neusten Versionen der Programme *Ghostscript* und *Ghostview* zum Download.
- [www.miktex.org](http://www.miktex.org): Die MikTeX-Homepage. Neben Dokumentationen, Links und Updates findet man hier auch die aktuellen MikTeX-Versionen.
- [www.winedt.com](http://www.winedt.com): Homepage des Editors *WinEdt*. WinEdt ist Shareware. Die Nutzung ist in den ersten Wochen frei, danach muss man sich (gegen eine angemessene Gebühr) registrieren lassen. Hier finden sich auch Wörterbuchdateien in verschiedenen Sprachen.
- [www.dante.de](http://www.dante.de): Homepage der „Deutschsprachigen Anwendervereinigung  $\TeX$ e.V.“. Links, FAQs, Dokumentationen und ein riesiges FTP-Verzeichnis mit Suchfunktion machen diese Seite zur ersten Anlaufstelle bei  $\TeX$ -Problemen und der Suche nach weiterer Software.
- [www.tug.org](http://www.tug.org): Die  $\TeX$ -User Group ist das englische Pendant zur Dante-Seite. Hier findet sich das Tool *epstopdf*.
- [www.ctan.org](http://www.ctan.org): Das „Comprehensive  $\TeX$ -Archive Network“ bietet einen riesigen FTP-Server, der u.a. auch von der Uni Bielefeld und anderen Institutionen gespiegelt wird, der alle nur erdenklichen Dateien zum Thema  $\TeX$  enthält.
- [www.imagemagick.org](http://www.imagemagick.org): ImageMagic konvertiert JPEG Files in EPS Dateien.
- [www.jam-software.com](http://www.jam-software.com): Homepage des *xl2LaTeX-Tools*, mit dem sich Microsoft EXCEL Tabellen als  $\LaTeX$ -Dateien abspeichern lassen.
- [www.wiwi.uni-bielefeld.de/~wolf](http://www.wiwi.uni-bielefeld.de/~wolf): Auf der Homepage von Peter Wolf finden sich die beiden Dokumente  *$\TeX$  in 7 Minuten* und *Weitere 7 Minuten für  $\TeX$* , die einen Schnellen Einstieg in die Syntax erlauben. Außerdem gibt es hier Hinweise zum Thema Literate Programming und REVWEB
- [www.winshell.de](http://www.winshell.de) Ein weiterer Freeware  $\LaTeX$ -Editor.
- [www.texniccenter.org](http://www.texniccenter.org) TeXNiXCenter ist ebenfalls eine Alternative zu WinEdt.

- [sunburn.stanford.edu/~knuth/](http://sunburn.stanford.edu/~knuth/) Die Homepage von Donald E. Knuth...
- Knuth, Donald E.: *Computers & Typesetting, Volume B: TeX: The Program*, Addison Wesley 1986. Mit diesem Buch fing alles an.
- Kopka, Helmut: *Latex - Einführung Band 1*, Addison Wesley 2000. Das Standardwerk zu L<sup>A</sup>T<sub>E</sub>X.