

Grazer Linuxtage 2004

T_EX your docs!

Stefan Sollerer

14. April 2004

1 Was ist LaTeX?

1.1 Die Geschichte von LaTeX

LaTeX ist ein Textsatzsystem, das von Autoren und Verlagen zum Verfassen von qualitativ hochwertigen Texten verwendet wird. Das Herz von LaTeX ist TeX, entwickelt von Donald E. Knuth. TeX beinhaltet ein paar wenige Anweisungen, die zum „Konstruieren“ eines Textes erforderlich sind. Brauchbar wird diese Ansammlung von Anweisungen erst durch LaTeX, das die TeX-Anweisungen zu leicht anzuwendenden Makros zusammenfaßt. LaTeX ist damit ein „Frontend“ für den eigentlichen Prozessor TeX.

Wie sind diese „Makros“ für uns anwendbar? Der Text ist wie beschrieben aus unter anderem aus Befehlen aufzubauen. Dabei werden diese Befehle in eine Textdatei geschrieben. Wie man diese Textdatei anlegt, ist gleichgültig – ob mit dem Lieblingseditor oder einer eigenen LaTeX-Entwicklungsumgebung. Diese Textdatei muß, um sie in ein fertiges Dokument umzuwandeln, nach Befehlen durchsucht werden. Dies geschieht durch das Programm LaTeX. Die angegebenen LaTeX-Makros werden in TeX-Anweisungen umgewandelt, das schlußendlich das Dokument mit entsprechenden Schriftgrößen und Abständen aufbaut. Mit einem Viewer wird das Ergebnis kontrolliert – je nachdem, ob man DVI, PS oder PDF als Endformat wählt, benötigt man einen geeigneten Viewer.

Die Umgebung bei der Arbeit mit LaTeX besteht also aus folgenden „Komponenten“:

- Editor
- LaTeX-Compiler
- Viewer (PDF, PS, DVI)

1.2 Warum ist für uns LaTeX interessant?

Wie der Name „Textsatzsystem“ bereits sagt, wird der Text gesetzt. Im Gegensatz zu „Textverarbeitungssystemen“, wo der eingegebene Text verarbeitet wird, ist die Einarbeitungszeit bei LaTeX sicher länger. Warum sollte man es sich also antun, so ein Textsatzprogramm zu erlernen? Der klare Vorteil liegt darin, daß man sich um die Formatierung keine Gedanken mehr machen muß.

Beim Anfertigen von Dokumenten spielt Typografie eine große Rolle. Typografie ist ein viele Teilbereiche umfassendes Themengebiet – allein Amazon bietet 180 Bücher zu Typografie und Layout an. Wo wird in Standarddokumenten Typografie sichtbar?

- Verhältnis der Schriftarten- und Größen bei Fließtext, Überschriften, Fußnoten, Bildbeschriftungen, ...
- Seitenaufbau, Satzspiegel, Seitenränder
- Abstände zwischen einzelnen Konstruktionselementen wie Fließtext, Überschriften, Aufzählungen

Zu Beginn fühlt man sich mit L^AT_EX oft eingesperrt, weil es „stur“ nach gewählter Dokumentart Schriftgrößenverhältnisse festlegt und eigene Formatierungsvorstellungen nicht so einfach zuläßt. Die Erfahrung hat jedoch gezeigt, daß das Gesamtbild eines fertigen Dokumentes umso besser aussieht, je mehr man sich auf die Formatierung eines Typografieprofis verläßt. Natürlich ist es andererseits auch möglich, die Formatierungsvorgaben des Verlages oder einem Institut zu erfüllen. Oft erhält man diese Vorgaben in Form einer Datei, die nach dem Einbinden die Formatierung des Dokumentes anpaßt.

Mit L^AT_EX erhält man auf einfache Weise ein gut austauschbares Format (PDF und/oder PostScript). Dabei können die Eigenschaften eines PDF-Dokumentes voll ausgenützt werden (Hyperlinks, Referenzen, Inhaltsangabe im Acrobat Reader, . . .).

1.3 Wofür ist L^AT_EX weniger geeignet?

Jedes Programm hat sein Einsatzgebiet und damit auch seine Grenzen. Bei L^AT_EX ist diese Grenze dann erreicht, wenn es um DTP (Desktop Publishing) geht. Natürlich wäre es möglich, mit L^AT_EX beispielsweise ein Plakat zu erstellen – DTP-Programme sind dafür jedoch besser geeignet. Mit diesen ist es einfacher möglich, ein Seitenlayout zu gestalten (Bilder absolut positionieren, Farbsättigungen einstellen, Farbanpassung von Schrift über einem Farbbild).

2 L^AT_EX lernen

Viele Studenten wissen, daß sie ihre Diplomarbeit mit L^AT_EX schreiben werden. Daß man dieses Programm erst lernen muß (wie jede andere Computeranwendung), erkennt man dann leider erst, wenn man versucht, die ersten Zeilen der Diplomarbeit zu verfassen. Je öfter man ein Programm verwendet, desto mehr bekommt man ein Gefühl dafür, erkennt Stärken und Schwächen, macht Fehler und lernt aus ihnen. Im Laufe des Studiums absolviert man immer wieder Lehrveranstaltungen, bei denen ein Bericht abzugeben ist. Solche Berichte sind sehr gut geeignet, um L^AT_EX kennen zu lernen. Nachdem man sich einmal mit der Grundstruktur einer TEX-Datei auseinandergesetzt hat, kann man sich schon an einen Laborbericht machen.

Der erste Schritt, L^AT_EX zu lernen, ist einfach:

2.1 Die Arbeitsumgebung installieren

Basissystem Man benötigt ein lauffähiges L^AT_EX-System mit dem enthaltenen Compiler und Paketen

- Linux: teTeX
- Windows: MikTeX

Editor Um die L^AT_EX-Datei zu erstellen, ist ein Editor erforderlich

- Kile

- Emacs
- Vim
- TeXnicCenter
- Winedt

Viewer Das Ergebnis kann mit einem Viewer betrachtet werden

- PDF: Acrobat, xpdf
- PS: gv
- DVI: xdvi, Yap

2.2 Editor öffnen und loslegen

Es gibt mehrere Möglichkeiten, die ersten Schritte mit L^AT_EX zu machen:

- Das Tutorial von L^AT_EX@TUG
- Man kann auch mit einem Buch und einem fertigen Dokument zum Analysieren beginnen
- Im Internet sind viele Einstiegstutorials zu finden
- L^AT_EX kann in der Gruppe erlernt werden
- Workshop bei den Grazer Linuxtagen 2004

3 Der Aufbau eines Dokumentes

datei.tex ist eine L^AT_EX-Datei. Wie bereits beschrieben, handelt es sich hierbei um eine Textdatei mit zwei Arten von Einträgen: Befehlen und Fließtext. Eine *.tex*-Datei ist nun im Allgemeinen folgendermaßen aufgebaut:

```
\documentclass[12pt]{article}

                                % Präambel
```

```
\begin{document}

                                % Inhalt
```

```
\end{document}
```

`\documentclass{article}` ist ein L^AT_EX-Befehl; solche Befehle beginnen mit `\` und enden meist mit einer zwingenden Angabe in geschwungenen Klammern. Wie im Beispiel können in eckigen Klammern Optionen angegeben werden. Mit `\documentclass{Dokumenttyp}` wird die **Art des Dokumentes** festgelegt.

- Standardklassen
 - article
 - report
 - book
- KOMA-Skripts
 - scrartcl
 - scrreprt
 - scrbook
 - scrlettr2
- ...

Bei den Standardklassen handelt es sich um die älteren Typen, die auch am meisten verbreitet sind. Die KOMA-Skripts sind auf europäische Layouts abgestimmt und mit Hilfe der Dokumentation sehr gut zu konfigurieren.

In der **Präambel** werden die Dokumenteigenschaften festgelegt. L^AT_EX kann beispielsweise um Pakete erweitert werden – vergleichbar mit „Plugins“. Kopf- und Fußzeilen, Dokumenttitel, Autor(en), Seitenränder werden ebenfalls in der Präambel festgelegt.

Neben den alleinstehenden Befehlen gibt es noch Umgebungen – diese werden in der TEX-Datei begrenzt durch

```
\begin{document}
```

```
\end{document}
```

Alles, was in der *document*-Umgebung steht, gehört zum **eigentlichen Dokument**. Dabei kann es sich um Fließtext handeln, aber auch um Befehle, Umgebungen oder dem mathematischen Modus.

4 Wie bringe ich L^AT_EX dazu, das zu machen, was ich will?

Wer sich daran macht, ein Dokument zu schreiben, hat eine konkrete Vorstellung vom Endprodukt. Während der Schreibarbeit tauchen immer wieder „Wünsche“ auf, wie der nächste Schritt aussehen soll. Nehmen wir an, ein Absatz wird fertig, und eine neue Überschrift soll eingebaut werden. Als erstes wird man sich Gedanken machen, welche Gliederungstiefe die Überschrift haben soll – eine Abschnittsüberschrift, ein neuer beschrifteter Paragraph, oder gar neues Kapitel. Anschließend sucht man den zugehörigen L^AT_EX-Befehl und fügt ihn in das Dokument ein.

Man kann sich vorstellen, daß es eine Unmenge an Befehlen gibt. Sich alle zu merken ist wohl ein Ding der Unmöglichkeit; für die üblichen Konstruktionselemente gibt es die Unterstützung des Editors, der auf Wunsch die Umgebung oder den Befehl einfügt.

Ich will ...

...eine Überschrift erstellen:

```
\section{Einleitung zur Diplomarbeit}

\subsection{Danksagung}
```

...eine Aufzählung erstellen:

```
\begin{itemize}
  \item der erste Punkt
  \item der nächste Punkt
\end{itemize}
```

...das Inhaltsverzeichnis einbinden: Aufgrund der befehlsartigen Angabe der im Dokument enthaltenen Verzeichnisse kann recht einfach eine Liste der Kapitelüberschriften mit zugehöriger Seitennummer erstellt und eingebunden werden. Dazu reicht ein einziger Befehl:

```
\tableofcontents
```

...eine Formel erstellen: L^AT_EX ist bekannt für seine Stärke, mit Formeln umzugehen. Formelumgebungen gibt es viele: beschriftet, unbeschriftet, ausgerichtet, mehrzeilig, usw. Eine einfache, beschriftete Formel wird durch folgende Umgebung erzeugt:

```
\begin{equation}

\end{equation}
```

Die Formeleingabe erfolgt dann in Pseudocode:

$$\sigma = \sqrt{a^2 + b^2}$$

wird erstellt mit

```
\sigma = \sqrt{a^2 + b^2}
```

...auf ein Kapitel verweisen : Referenzen bestehen aus zwei Komponenten – der Beschriftung und dem Verweis darauf. Um beispielsweise auf den Abschnitt *Versuchs-anordnung* zu verweisen, muß der Code folgendermaßen aussehen:

```
\section{Versuchsanordnung}
\label{sec:versuch}
```

In Kapitel `\ref{sec:versuch}` wird der Versuch erklärt.

Dies würde ergeben:

In Kapitel 3 wird der Versuch erklärt.

- ... **Bilder** einbinden: bei längeren Arbeiten können Bilder den Lesefluß stören, wenn sie in der Mitte der Seite platziert werden. Je nach Größe der Grafik kann es geschehen, daß diese auf die nächste Seite verschoben werden muß, und dadurch auf der vorherigen Seite ein Freiraum entsteht. Deshalb werden Abbildungen oft an den Beginn oder das Ende einer Seite verschoben.

In L^AT_EX stehen sogenannte „Gleitumgebungen“ zur Verfügung. Im Code wird das Bild in einer Gleitumgebung an der Stelle eingefügt, an der es ungefähr platziert werden soll, L^AT_EX verschiebt die Abbildung dann je nach Wunsch an einen Seitenkopf oder ein Seitenende. Auf die verschobene Grafik kann mit Referenzierungselementen verwiesen werden (*In Abbildung 3.4 auf Seite 23 ...*).

```
\begin{figure}[tb]
  Bild hier einfügen
  \caption{Bildbeschriftung}
  \label{fig:referenzierungsname}
\end{figure}
```

Diese Methode der Platzierung von „groben“ Objekten kann auch bei Tabellen angewendet werden; hier wird die Beschriftung dementsprechend angepaßt (*In Tabelle 3.4 auf Seite 23 ...*).

```
\begin{table}[tb]
  Tabelle hier einfügen
  \caption{Tabellenbeschriftung}
  \label{fig:referenzierungsname}
\end{table}
```

- ... **Programmiercode** einbinden: gerade für Programmierer interessant ist die Möglichkeit, Programmiercode zu dokumentieren. Code kann dabei 1 : 1 per Copy & Paste in das L^AT_EX-Dokument eingebunden werden. Nach Angabe der Programmiersprache wird der Code entsprechend formatiert. Das beste Beispiel sind die in dieses Dokument eingebundenen L^AT_EX-Codeschnipsel.

- ... **auf Literatur verweisen:** BiB_TE_X ist für die Verwaltung von Literatur und Verweise auf diese verantwortlich. Die Vorgehensweise ist eigentlich recht einfach:

1. In einer externen Datei werden Informationen zu der verwendeten Literatur angelegt:

```
@Article{Sollerer04,  
  author =      {Stefan Sollerer},  
  title =       {TeX your docs!},  
  journal =     {Grazer Linuxtage 2004},  
  year =        {2004},  
}
```

2. Das Hauptdokument enthält 2 Einträge bezüglich der Bibliographie:
 - der Name der externen Literaturdatei
 - der zur Formatierung des Verzeichnisses zu verwendende Stil
3. „Die Angabe stammt aus `\cite{Sollerer04}`“ wird damit zu „Die Angabe stammt aus [1]“; der ausgewählte Artikel wird in das Literaturverzeichnis übernommen und entsprechend formatiert.

5 Ausblick – was kann L^AT_EX sonst noch?

- Serienbriefe verfassen und dabei auf eine Datenbank zugreifen
- MusicT_EX
- Präsentationen