

Einführung in das $\text{T}_{\text{E}}\text{X}$ –TPP5.01–Paket

Wie arbeiten diese beiden Programme
einzeln und gemeinsam

TPP ist zunächst ein normales Textverarbeitungsprogramm. Es besitzt kein WYSIWYG. Dafür lassen sich beim Ausdruck alle Zeilen durchnumerieren. Dennoch bietet es als Textprogramm alle üblicherweise benötigten Featurs. Ich benutze es ausschließlich. Es fehlt nichts. — Ohne seine guten Eigenschaften als Textprogramm irgendwie einzuschränken, zielt es andererseits auf $\text{T}_{\text{E}}\text{X}$. Nach der Umsetzung des Textes kann man das von $\text{T}_{\text{E}}\text{X}$ organisierte(!) Schriftbild am Monitor betrachten und ggf. korrigieren. Und dieses Schriftbild ist hervorragend: Es entspricht absolut dem Buchdruck! Besonders wenn man einen hochauflösenden Monitor benutzt, er ist aber keinesfalls Voraussetzung. Das Druckergebnis wird immer überzeugen; denn $\text{T}_{\text{E}}\text{X}$ –Dateien könnten ohne weiteres als nicht mehr zu korrigierende Druckvorlagen dienen! Von einer Reihe von Verlagen wird $\text{T}_{\text{E}}\text{X}$ deshalb auch eingesetzt. — $\text{P}_{\text{a}}\text{T}_{\text{E}}\text{X}$ ist die Umsetzung von $\text{T}_{\text{E}}\text{X}$ und $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ auf den Amiga.

$\text{T}_{\text{E}}\text{X}$ übernimmt nach dem Formulieren des Textes die Arbeit des Schriftsetzers aus *alten Zeiten*. Heute erledigen dessen Arbeit fast überall Computer der verschiedensten Art. Der Schriftsetzer bekam als Druckvorlage ein Manuskript, das auch alle Hinweise darüber enthielt, wie das Druckerzeugnis später aussehen sollte: Schrifttypen, Schriftgrößen für Überschriften, Texte und Fußnoten, Hervorhebungen durch kursiv oder gesperrt usw. usw. Der Schriftsetzer hatte diese Anweisungen auszuführen und zusätzlich dafür zu sorgen, daß das Druckbild eine einheitliches Aussehen zeigt. Im Zeitungsdruck dagegen finden wir heute nicht selten längere Lücken zwischen einzelnen Wörtern und auch den Buchstaben, um (kurze) Zeilen im Blocksatz (schnell!) setzen zu können. Der Schriftsetzer hingegen variierte die (sehr kleinen) Abstände zwischen den Buchstaben innerhalb der Wörter und die größeren zwischen denselben, bis der Eindruck eines gleichmäßigen Aussehens erreicht war. Weiterhin variierte er ggf. auch die Absatzabstände und den Seitenumbruch, damit nicht ein neuer Absatz auf der letzten Zeile einer Seite beginnt bzw. er mit seiner letzten Zeile auf einer neuen Seite endet. Besondere Sorgfalt wurde von ihm beim Setzen von Titelseiten, Urkunden usw. verlangt.

Das zu automatisieren, das ist die Aufgabe der verschiedensten Programme, die heute im Verlagswesen eingesetzt werden. Ein solches Programm, das der Autor selbst nutzen kann, ist $\text{T}_{\text{E}}\text{X}$. Eine bestimmte Notwendigkeit für Autoren besteht durchaus, denn eine Reihe von Verlagen *wünschen*, daß ihre Autoren schon eine relativ fertige Druckvorlage einreichen und dafür z.B. $\text{T}_{\text{E}}\text{X}$ einsetzen sollten, wenn es ihnen irgendwie zur Verfügung steht. An Hochschulen und in wissenschaftlichen Instituten ist dies häufig der Fall. $\text{T}_{\text{E}}\text{X}$ liefert bei sorgfältiger Arbeit nämlich ein fast druckreifes Ausgabefile (.dvi), das von allen Computersystemen gelesen werden kann, wenn auf ihnen ebenfalls die ihrem System entsprechende $\text{T}_{\text{E}}\text{X}$ –Implementation installiert ist.

Der *Entwickler* von $\text{T}_{\text{E}}\text{X}$, der amerikanische Professor D.E. KNUTH hat damit eine bedeutende Pionierarbeit geleistet, die er außerdem auch noch der Allgemeinheit kostenlos, also als PD zur Verfügung stellte. Davon darf man aber keinesfalls ableiten, $\text{T}_{\text{E}}\text{X}$ sei ein Produkt minderer Qualität. $\text{T}_{\text{E}}\text{X}$ ist vielmehr eine Meisterleistung. Seine hohe Bedeutung kommt z.B. auch darin zum Ausdruck, daß es weltweit einige Gremien gibt, die sich mit diesem Programm befassen. In Deutschland ist dies DANTE, die Deutschsprachige Anwendervereinigung TeX e. V. in Heidelberg. Dort kann man sich ggf. auch Hilfe holen. Und es gibt eine ganze Reihe von *dicken* Büchern, die sich mit $\text{T}_{\text{E}}\text{X}$, $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ und *META*FONT (s.u.) befassen.

$\text{T}_{\text{E}}\text{X}$ –Texte werden in der Regel zunächst mit einem beliebigen Editor geschrieben und danach über die 'shell' der $\text{T}_{\text{E}}\text{X}$ –Bearbeitung (entspricht einer Compilierung) zugeführt. $\text{T}_{\text{E}}\text{X}$ untergliedert den Text in die einzelnen Absätze. Grob gesagt und bildlich gesprochen beginnt $\text{T}_{\text{E}}\text{X}$ jetzt, jeden Absatz *so lange zu schütteln*, bis einerseits die Abstände der Buchstaben im einzelnen Wort und andererseits die Abstände zwischen den Worten weitgehend gleich sind und gleichzeitig ein sauberer Blocksatz mit automatischer und richtiger (aber möglichst seltener!) Silbentrennung entsteht. Da aber eine Seite (oder aber auch alle Seiten eines Buches) ein gleiches Aussehen haben sollen, müssen schließlich diese Abstände über alle Absätze angeglichen werden usw. usw.

T_EX löst diese Aufgabe dadurch, daß jedes Zeichen entsprechend seiner individuellen Breite eine ihn umrahmende (unsichtbare) Box erhält. Diese Box ist (abgeleitet von der Setztechnik des Schriftsetzers) elastisch, d.h. sie ist um je einen bestimmten Betrag komprimier- bzw. dehnbar. Aus diesen Boxen der Buchstaben entstehen Boxen für Wörter, Zeilen, Absätze und schließlich Seiten. Dabei werden die Wort- und ggf. auch die Zeilen- und Absatzabstände in bestimmten Grenzen variiert. Zu große oder zu enge Abstände würden zu keinem guten Aussehen führen!

Daß dieses Problem der Computer nicht immer zu *seiner* Zufriedenheit lösen kann, gibt er beim Compilieren durch entsprechende Warnungen zu erkennen. Entweder man akzeptiert diese oft sehr minimalen Abweichungen oder man stellt den Text etwas um und compiliert erneut. Man kann dem Text das Schlüsselwort 'Salopp' voranstellen, dann nimmt er es nicht so genau, z.B. für nicht so wichtige Dokumente.

In speziellen Druckerzeugnissen sind **Hervorhebungen der verschiedensten Art** nötig. *Das geht vom einfachen ZEILENWECHSEL* (nicht am Ende einer Zeile) über das Einfügen von Tabellen und Listen (beides geht über TPP besonders leicht!) bis zur Darstellung komplizierter mathematischer Formeln. Ein längeres Druckerzeugnis (z.B. Artikel oder Buch) benötigen zusätzlich (meist) ein Titelblatt, ein Inhalts-, Stichwort- und Literaturverzeichnis mit entsprechenden Seiten- und auch Querverweisen und auch einer Fußnotenverwaltung.

Dieses zu realisieren, übernimmt (unter TPP) L^AT_EX vollautomatisch. Das Programmpaket L^AT_EX baut auf T_EX auf und schafft eine benutzerfreundliche Ebene. L^AT_EX entwickelte der Amerikaner L. LAMPORT. Dazu schuf er eine sehr große Reihe von Befehlen, die die Nutzung von T_EX sehr wesentlich vereinfachte. Insbesondere schuf er *Rahmenprogramme* mit deren Hilfe man bestimmte Schriftstücke wie Briefe, Geschäftsbriefe, Artikel, Reviews und Bücher druckfertig schreiben kann, ohne sich mit T_EX beschäftigen zu müssen. Theoretisch könnten die so beschriebene Disks sofort der Druckerei übergeben werden. Der Setzer ist arbeitslos.

Für *ausgefallene* Druckerzeugnisse wie z.B. für die Werbung mit flippiger Schrift und bunten Bildern, für Festzeitungen oder besonders gestaltete Urkunden ist T_EX weniger geeignet. Dazu ist T_EX nicht konzipiert worden.

T_EX stammt aus den USA und legt zunächst das englische Alphabet zugrunde. D.h. T_EX kennt primär nicht die deutschen Umlaute und das 'ß' bzw. die speziellen Buchstaben anderer Sprachen. Diese müssen durch entsprechende Befehle T_EX mitgeteilt werden. Anstelle z.B. des Umlautes ü müßte man schreiben `\“{u}` oder `\“u`. (Die geschweiften Klammern haben hier eine gewisse *Schutzfunktion*.) Besonders solche zusätzlichen Befehle stören natürlich den normalen gewohnten Schreibfluß: Wendet man allein das T_EX-Paket z.B. auf der Pearls an (Es nennt sich hier PasT_EX und ist eine PD-Umsetzung auf dem Amiga von Georg Heßmann: Ihm tausend Dank für solch eine Sisyphe-Arbeit!), dann müßte man alle Umlaute wie eben angegeben schreiben.

Aber wir haben ja 'T_EXt Plus Professional 5.01', das uns von der Anwendung der wichtigsten T_EX- und L^AT_EX-Befehle befreit. (Eben wurde im unter TPP geschriebenen Text nur der Größenbefehl für das Wort 'Aber' gegeben. Den Zeilenabstand variierte T_EX selbst!) Während auf anderen Computersystemen beim Einsatz von T_EX der Text noch programmiert werden muß, L^AT_EX eine starke Entlastung bringt, macht TPP uns das alles fast kinerleicht.

Mit TPP5.01 holt man sich ein vorbereitetes Formblatt auf den Monitor und füllt es einfach aus. Dabei braucht man nur die normalen Schreibregeln zu beachten: Wörter werden durch (beliebig viele) Spaces getrennt. Ein Zeilenwechsel zählt auch als Space. Mehrere Zeilenwechsel trennen Abschnitte von einander. Spaces am Anfang einer Seite oder Zeile werden nicht beachtet. (Sollen sie beachtet werden, dann gibt es dazu spezielle Befehle.)

Man lade sich dafür über die Menüleiste 'Dokument' -> 'Lade Configuration'

`TPP_small.config`.

Damit wird TPP von der normalen Textverarbeitung auf $\text{T}_{\text{E}}\text{X}$ umgestellt. Danach hole man sich folgendes File herein:

`TPP:Example/deutsch/BriefFormblatt.tpp` .

Man speichere es sofort unter einem anderen Namen ab, denn das UrsprungsFormblatt muß ja erhalten bleiben! (Später sollte man das leere Formblatt, evtl. an seine speziellen Bedürfnisse angepaßt, an einer bequemen Stelle ablegen.)

In 'Schriftfabe 1' (z.B. weiß) erscheinen die Schlüsselwörter. Zwischen diese schreibt man, so wie vorgegeben, seine Anschrift, Absender usw. und schließlich den gewünschten Text. Nicht jedes Schlüsselwort *muß* ausgefüllt werden. Über die Menüleiste 'Dokument' -> 'Vorschau' leitet man die Übersetzung und Compilierung ein.

Zunächst speichert TPP den Text automatisch ab. Danach übersetzt es die Schlüsselwörter in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Befehle und schafft dadurch ein `.tex`-File. (Das sonst übliche Ausgangsfile für $\text{T}_{\text{E}}\text{X}$ unter $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$. Man sollte sich so ein `.tex`-File einmal ansehen!) Dieses wird anschließend von $\text{P}^{\text{a}}\text{s}\text{T}_{\text{E}}\text{X}$ übernommen und zum `.dvi`-File compiliert (und dabei *durchgeschüttelt*). Bei der Auswahl von 'Vorschau' wird der geschriebene Brief anschließend so angezeigt, wie er mit $\text{T}_{\text{E}}\text{X}$ auch ausgedruckt werden kann. Hätte man statt dessen 'Compilieren' ausgewählt, wäre allein die Anzeige nicht erfolgt. Mit dem Menüpunkt 'Drucken' wird der eingelegte Bogen formgerecht im Grafikdruck ausgegeben. Vorher muß für den Drucker einmalig die richtige Einstellung gefunden werden. $\text{T}_{\text{E}}\text{X}$ bietet Treiber für die verschiedensten Druckersysteme an. Wo ist das nicht so?! (s. Datei `TeXmf/config/dviprint.printer`)

Das Beispiel zeigt, daß unter TPP kein Programmieren der $\text{T}_{\text{E}}\text{X}$ -Texte mehr notwendig ist. Will man (später!) weitere Gestaltungselemente nutzen, dann kommt man um den einen oder anderen $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Befehl nicht herum. Dennoch bleibt $\text{T}_{\text{E}}\text{X}$ unter TPP kinderleicht zu bedienen und die Ergebnisse, einen ausreichend guten Drucker vorausgesetzt, halten jeder Kritik stand. Die langweilige (meist nicht proportionale) übliche Computerschrift will man dann nicht mehr sehen!

TPP stellt uns eine komfortable Oberfläche für $\text{T}_{\text{E}}\text{X}$ zur Verfügung. Sogar mir, dem Autor dieses Beitrages, sind die ohne TPP unbedingt zu beherrschenden Shell-Befehle zur Bedienung von $\text{P}^{\text{a}}\text{s}\text{T}_{\text{E}}\text{X}$ Fremdworte geblieben!

Für einen Brief müssen die folgenden fünf Schlüsselwörter unbedingt benutzt werden: Brief, Absender, Adresse, Beginn, Ende. Sie müssen aber nicht alle ausgefüllt werden! Die anderen Schlüsselwörter sind wahlweise verwendbar. Somit kann man auch einen mehr privaten Brief ohne die geschäftlichen Attribute schreiben. Beim formulierenden Schreiben eines Textes braucht man nicht auf ein schönes Aussehen des Schriftbildes zu achten (s. dazu den Beispielbrief). Man kann sich absolut auf den Inhalt seines Textes konzentrieren, kann *wild* ändern, wenn man nur die Leerzeilen zur Absatztrennung beachtet. Wichtig ist, daß die meisten TPP-Schlüsselwörter am Zeilenanfang stehen müssen! Wenn man später ein Filing für $\text{T}_{\text{E}}\text{X}$ erworben hat, dann kann man zur Gestaltung weitere $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Befehle anwenden, die man dazu unter TPP:Parse eintragen oder sich merken und original an den entsprechenden Textstellen einfügen kann (s. dazu im Beispielbrief).

Will man eine kurze Notiz unter $\text{T}_{\text{E}}\text{X}$ schreiben, dann setze man die fünf notwendigen Schlüsselwörter für einen Brief in 'Farbe 1' ohne Zwischentexte untereinander. Zwischen die Schlüsselwörter Beginn und Ende setze man seinen Text und compiliere das ganze. Fertig!

Zusätzliche Schlüsselwörter findet man in den Requestern unter dem TPP-Menü-Punkt 'TeX'. Dazu bitte die bearbeitete Bediananleitung 'TPP_Dok_bearb.' beachten.

In einigen seltenen Fällen kann es vorkommen, daß diese Schlüsselwörter nicht ausreichen. Dann kann man in den TPP-Text originale $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Befehle einsetzen. Dazu studiere man die einschlägigen Doks in $\text{P}^{\text{a}}\text{s}\text{T}_{\text{E}}\text{X}$. Leider sind dort die benötigten Doks nicht immer schnell zu finden. Es fehlt ein Leitfaden, der dem Newcomer hilft die richtige Reihenfolge der zu lesenden Doks vorzuschlagen. (Vielleicht findet der Autor dieser Zeilen irgendwann mal die Zeit, Abhilfe zu schaffen.)

Einige solcher häufig benötigten \LaTeX -Befehle finden sich im Dok-Kapitel \TeX -Tips. Jeder \LaTeX -Befehl beginnt mit einem Backslash '\'. In der Regel werden die Befehle klein geschrieben. \LaTeX unterscheidet zwischen Groß- und Kleinschreibung! Befehlszusätze sind in Klammern verschiedener Art eingeschlossen. Verschachtelungen von Klammerausdrücken sind bei bestimmten Befehlen möglich. Und: Jeder \LaTeX -Befehle endet mit einem 'Nicht-Buchstaben' oder einer 'Nicht-Zahl', also z.B. mit Punkt, Komma, Space oder Klammer. Das hat zur Folge, daß nach einem durch einen \LaTeX -Befehl erzeugten Ausdruck kein Space erfolgt. Diesen muß man durch einen `\dirket(!)` hinter den Befehl gesetzten \LaTeX -Space (= Backslash und eine Leertaste) erzeugen, z.B. '`\LaTeX__`'

TPP verarbeitete anfangs \LaTeX -Befehle nicht so ohne weiteres, deshalb findet man in der Original-Dok noch die *Einrahmung* dieser Befehle durch die Schlüsselwörter 'TeX' und 'Ende'. Diese Einrahmung ist aber jetzt nicht mehr nötig. TPP5.01 verarbeitet jetzt fast jeden originalen \LaTeX -Befehl direkt. Nur in Ausnahmefällen und für den mathematischen Modus werden 'TeX' und 'Ende' noch benötigt.

\LaTeX bietet u.a. verschiedene Schrifttypen in verschiedenen Größen. Ist eine bestimmte Größe nicht verfügbar, dann können sie automatisch vom Programmteil *METAFONT* erzeugt werden.

\TeX zusammen mit \LaTeX bietet eine fast vollständige Menge an Befehlen, um jedes (*seriöse*) Druckerzeugnis gestalten zu können. Um die Möglichkeiten abzurufen, sollen einige mathematische *Phantasie*-Formeln gezeigt werden. Um deren Schreibweise unter TPP zu sehen, lade man sich das File 'EinfTeX.tpp'. Das einfache TPP-Briefformat, das für diese Einführung gewählt wurde, verarbeitet mathematische Formeln nicht. Deshalb mußten sie im TPP-File, wie gerade erwähnt, um die Schlüsselwörter 'TeX' und 'Ende' ergänzt werden.

$$S(z) = -\cos\left(\frac{\pi}{2}z^2\right) \sum_{n=0}^{\infty} \frac{(-1)^n \pi^{2n+1}}{1 \cdot 3 \cdots (4n+3)} z^{4n+3} + \sin\left(\frac{\pi}{2}z^2\right) \sum_{n=0}^{\infty} \frac{(-1)^n \pi^{2n}}{1 \cdot 3 \cdots (4n+1)} z^{4n+1}$$

$$\sqrt[n]{\text{frac}x^n - y^n 1 + u^{2n}}$$

$$\sqrt[3]{-q + \sqrt{q^2 + p^3}}$$

$$A(\mathcal{Q}) = \sum_{\substack{(1,2,\dots,n) \\ p_1 < p_2 < \dots < p_{n-k}}} \Delta_{\substack{p_1 p_2 \dots p_{n-k} \\ p_1 p_2 \dots p_{n-k}}} \sum_{q_1 < q_2 < \dots < q_k} \begin{vmatrix} a_{q_1 q_1} & a_{q_1 q_2} & \cdots & a_{q_1 q_k} \\ a_{q_2 q_1} & a_{q_2 q_2} & \cdots & a_{q_2 q_k} \\ \dots & \dots & \dots & \dots \\ a_{q_k q_1} & a_{q_k q_2} & \cdots & a_{q_k q_k} \end{vmatrix}$$

und noch einige Zeichen aus dem *unerschöpflichen* Vorrat:

$$\prod \int \oint \cup \lim_{x \rightarrow \infty} \flat \# \natural \Re \cup$$

$$\clubsuit \heartsuit \perp \neq \updownarrow \Gamma \gamma \Xi$$

Schließlich hat TPP+T_EX noch einen weiteren großen großen Vorteil: Man kann vor dem ersten Schlüsselwort (z.B. 'Brief') bzw. auch nach 'Ende' beliebigen Text schreiben, der nicht mit in das Dokument gelangt. Gleiches gelingt für Zeilen mitten im Text, die mit dem Zeichen '%' beginnen. Somit können alle für den Autor wichtigen eignen Bemerkungen an Ort und Stelle angebracht werden, ohne daß sie der Empfänger zu Gesicht bekommt.

Vor dem ersten Schlüsselwort lassen sich auf diese Weise günstig andere Schlüsselwörter und originale L^AT_EX-Befehle als 'Spickzettel' niederlegen und bei Bedarf schnell an den Schreibort holen. Das geht mithin schneller als über den Menüpunkt 'TeX'.

Das war's! Ich hoffe, diese Einführung hat den einen oder anderen auf die Möglichkeiten von T_EX und L^AT_EX neugierig gemacht. Weitere (konkretere) Informationen bietet die überarbeitete Dok zu TPP. Wer noch mehr wissen möchte, dem sei u.a. die deutschsprachige Literatur von Kopka empfohlen, s. TPP-Dok.

Der Bearbeiter der Dok zu TPP von Martin Steppler ist:

Christian Scharrnbeck

R.-Schumann-Str. 21

08280 Aue

Aus Telecom-Gründen noch keine e-Mail!

Aue, den 3.Juni 1996