

Kapitel 1

Einführung

1.1 Warum L ^A T _E X und nicht die Textverarbeitung xy?	2
1.2 Warum LuaL ^A T _E X und nicht pdfL ^A T _E X?	7
1.3 Das Prinzip	9
1.4 Externe L ^A T _E X-Anwendungen	10

Die formale Ausführung einer wissenschaftlichen Arbeit kann man in zahlreichen Publikationen nachlesen, beispielsweise in Balzert u. a. (2008), Toraman (2016) und Wyrzens u. a. (2012). Daneben gibt es dann noch diverse Veröffentlichungen, die ein bestimmtes Textsystem als Voraussetzung für die Erstellung der Arbeit haben, beispielsweise Farke (1997), Koch (2010) und Seimert (2011). Für den Einsatz von L^AT_EX zur Erstellung einer wissenschaftlichen Arbeit gibt es unter anderem die Veröffentlichungen von Machert (2013), J. Schlosser (2016), P. Schlosser u. a. (2007) und Würth u. a. (2018), wobei nur Würth u. a. (2018) als aktuell bezeichnet werden kann. Mit der hier vorliegenden Arbeit wird konsequent auf die neuesten Entwicklungen von T_EX gesetzt: Als Standardprogramm wird LuaT_EX verwendet, als Dokumentenklasse KOMA-Script und als Bibliografieprogramm Biber mit dem korrespondierenden BibL^AT_EX.

Die verwendete T_EX-Distribution war T_EXLive (<http://www.tug.org/texlive/>), was aber im Prinzip für die eigentliche Arbeit mit L^AT_EX unerheblich ist; mit der Distribution MiK_TE_X (<http://www.miktex.org>) wird dasselbe Ergebnis erreicht.¹ Kommerzielle Distributionen werden hier nicht behandelt.



¹Die Distribution MacT_EX ist eine spezielle Anpassung an T_EXLive und *proT_EX* eine spezielle Version von MiK_TE_X.

1.1 Warum L^AT_EX und nicht die Textverarbeitung xy?

Die Anwendung von L^AT_EX zur Erstellung einer wissenschaftlichen Arbeit kann man für den Bereich Mathematik als »flächendeckend« bezeichnen. Es besteht aber heutzutage eigentlich kein direkter Zusammenhang mehr zwischen Mathematik und L^AT_EX. Auch für Sprachwissenschaftler, die beispielsweise linguistische Bäume erstellen müssen oder andere Autoren, die sehr große Dokumente erstellen wollen, haben durch die Anwendung von L^AT_EX viele Vorteile. Im folgenden Abschnitt wird gezeigt, dass auch eine relativ kurze wissenschaftliche Arbeit durch die Anwendung von L^AT_EX mindestens ästhetische Vorteile gegenüber den allgemein bekannten Textverarbeitungsprogrammen hat.

Die folgende Abbildung 1.1 zeigt den typischen Umbruch einer Textverarbeitung. Um den Unterschied zu einem Satzprogramm zu verdeutlichen, wurde die Trennung von Wörtern absichtlich nicht zugelassen. Textverarbeitungen beziehen den Umbruch auf eine einzelne Zeile, indem diese beendet wird, wenn ein weiteres Wort nicht mehr in diese passt. Dies führt im folgenden Beispiel dazu, dass die dritte Zeile mit einem sehr großen Wortabstand gesetzt wird, was optisch immer sehr zweifelhaft erscheint.

Neithardt von Gneisenau, der Kommandant der Festung Kolberg, deren ruhmreiche Verteidigung einen der interessantesten Abschnitte dieses Buches (von <http://www.gutenberg.org/files/23333/23333-8.txt>) bildet, 1760 geboren, hatte schon in einem zu Erfurt garnisonierenden österreichischen und danach in einem der Regimenter des Markgrafen von Ansbach-Bayreuth gedient, die in englischem Solde in und gegen Amerika kämpften, als Friedrich der Große ihn 1786 als Premierleutnant in die preußische Armee aufnahm.

Abbildung 1.1: Zeilenumbruch einer Textverarbeitung (LibreOffice).

Bei einer Textverarbeitung erfolgt *nach* einem Zeilenumbruch unabhängig von den zukünftigen Zeilenumbrüchen keine Änderung mehr. Diese Tatsache führt immer dann zu ungünstigen Umbrüchen, wenn die Zeile Bestandteile enthält, die nicht getrennt werden sollen oder können. In dem angegebenen Beispiel ist es eine lange Internetadresse, die ohne externen Eingriff nicht getrennt wird. Vergleicht man dazu Abbildung 1.2, welches die Ausgabe von Lua_T_EX darstellt, so fällt sofort auf, dass sich die Größe der Lücken verringert hat.

Neithardt von Gneisenau, der Kommandant der Festung Kolberg, deren ruhmreiche Verteidigung einen der interessantesten Abschnitte dieses Buches (von <http://www.gutenberg.org/files/23333/23333-8.txt>) bildet, 1760 geboren, hatte schon in einem zu Erfurt garnisonierenden österreichischen und danach in einem der Regimenter des Markgrafen von Ansbach-Bayreuth gedient, die in englischem Solde in und gegen Amerika kämpften, als Friedrich der Große ihn 1786 als Premierleutnant in die preußische Armee aufnahm.

Abbildung 1.2: Absatzumbruch eines Satzprogramms (Lua_T_EX).

Das Programm LuaT_EX ist keine Textverarbeitung im eigentlichen Sinne. Dies wird schon daran deutlich, dass es völlig unerheblich ist, mit welchem System der Quelltext für das Programm LuaT_EX erstellt wird. Der Anwender programmiert, d. h., er erstellt einen Programmcode, welcher als Ausgabe einen formatierten Text erzeugt. Es erleichtert Anfängern den Umgang mit dem Gesamtsystem, wenn man sich auch als (*Text*-)Programmierer versteht. Dies bedeutet, dass es völlig unwichtig ist, welche Formatierung der Quelltext aufweist, solange man die Regeln der dem System zugrundeliegenden Befehlssyntax beachtet. Dies betrifft primär den Zeilenumbruch; ein Satzprogramm umbricht grundsätzlich Absätze in Zeilen, wohingegen alle Textverarbeitungsprogramme nur einzelne Zeilen umbrechen.

Da bei LuaT_EX *immer* der gesamte Absatz für den Umbruch der einzelnen Zeilen herangezogen wird, kann das Aufeinandertreffen von sehr eng und sehr weit gesetzten Zeilen vermieden werden; jeder Umbruch einer Zeile steht in Bezug zu den vorhergehenden und nachfolgenden Zeilen. Vereinfacht formuliert kann man sagen, dass LuaT_EX vier optische Kriterien für den Satz von Zeilen kennt: *eng*, *weniger eng*, *weit* und *sehr weit*. Absatz-
umbruch

Zeilen können

- eng gesetzt sein:
Dies ist eine sinnlose Zeile mit etwas Text
- weniger eng gesetzt sein:
Dies ist eine sinnlose Zeile mit etwas Text
- weit gesetzt sein:
Dies ist eine sinnlose Zeile mit etwas Text
- sehr weit gesetzt sein:
Dies ist eine sinnlose Zeile mit etwas Text

Umbruchkriterium

Für jeden Absatz gilt: Zwei aufeinander folgende Zeilen müssen *gleiche* oder *benachbarte* Formatierungskriterien erfüllen!

Dadurch ist es bei LuaT_EX unmöglich, dass eine sehr weit gesetzte Zeile einer eng gesetzten folgt, was ansonsten zu den unangenehmen Lücken aus Abbildung 1.1 auf der vorherigen Seite führt. Erlaubt man zusätzlich Worttrennungen einschließlich eines automatischen Umbruchs der Internetadresse, so kann das optische Bild bei LibreOffice nur etwas verbessert werden (Abbildung 1.3). Erst bei der Anwendung von LuaT_EX ergibt sich mit Abbildung 1.4 auf der nächsten Seite ein ansprechendes Ergebnis.

Neithardt von Gneisenau, der Kommandant der Festung Kolberg, deren ruhmreiche Verteidigung einen der interessantesten Abschnitte dieses Buches (von <http://www.gutenberg.org/files/23333/23333-8.txt>) bildet, 1760 geboren, hatte schon in einem zu Erfurt garnisonierenden österreichischen und danach in einem der Regimenter des Markgrafen von Ansbach-Bayreuth gedient, die in englischem Solde in und gegen Amerika kämpften, als Friedrich der Große ihn 1786 als Premierleutnant in die preußische Armee aufnahm.

Abbildung 1.3: Zeilenumbruch mit Worttrennungen (LibreOffice).

Neithardt von Gneisenau, der Kommandant der Festung Kolberg, deren ruhmreiche Verteidigung einen der interessantesten Abschnitte dieses Buches (von <http://www.gutenberg.org/files/23333/23333-8.txt>) bildet, 1760 geboren, hatte schon in einem zu Erfurt garnisonierenden österreichischen und danach in einem der Regimenter des Markgrafen von Ansbach-Bayreuth gedient, die in englischem Solde in und gegen Amerika kämpften, als Friedrich der Große ihn 1786 als Premierleutnant in die preußische Armee aufnahm.

Abbildung 1.4: Absatzumbruch mit Trennungen und Linkumbruch (L^AT_EX).

Etwas Theorie

Der Umbruch für den Absatz wird nach dem sogenannten Best-Fit-Algorithmus vorgenommen; von allen möglich Umbrüchen eines Absatzes wird letztlich derjenige ausgewählt, der dem idealen Umbruch am nächsten kommt. Der ideale Umbruch hat zwischen allen Wörtern denselben minimalen Abstand und kommt ohne Trennungen aus. Lua_T_EX geht hierbei in maximal drei Stufen vor:

1. Versuch, einen optimalen Umbruch ohne Trennungen zu erhalten.
 - Genügt das Ergebnis den Vorgaben, wird der Umbruch akzeptiert und als abgeschlossen betrachtet.
 - In der T_EX-Terminologie bedeutet dies, dass die Dehnungspunkte (`\badness` – Abstände zwischen den Wörtern) *jeder* umbrochenen Zeile den Wert von `\pretolerance=100` nicht überschreiten.
 - Wegen nicht vorhandener Trennstellen benötigt der 1. Durchgang sehr wenig Rechenzeit!
2. Versuch, einen optimalen Umbruch mit Trennungen zu erhalten.
 - Es kommen als sprachenspezifische Trennstellen die Textstellen in Frage, die Lua_T_EX durch seinen Trennalgorithmus findet.
 - Der Umbruch wird akzeptiert, falls alternativ
 - die Dehnungspunkte (`\badness`) jeder umbrochenen Zeile den Wert von `\tolerance=200` nicht überschreiten und `\emergencystretch` (hier gleich 30.0pt) einen positiven Wert aufweist;
 - falls `\tolerance` den Wert 10000 hat; dann kommen zusätzlich `\hfuzz=0.1pt` und `\hbadness=1000` zum Tragen.
3. und letzter Versuch, durch zusätzliche Zwischenräume.
 - Der intern vorgegebene oder vom Anwender geänderte Wert von `\emergencystretch` (30.0pt) wird zur erlaubten Dehnungsmöglichkeit jeder Zeile addiert (siehe auch Abschnitt 6.13.4 auf Seite 117).
 - Durch diese Addition werden die zulässigen Dehnungspunkte jeder Zeile verringert.
 - Abschließende Entscheidung mithilfe der Register `\hfuzz=0.1pt` und `\hbadness=1000`.

Hierbei wird, wie oben erwähnt, die Berechnung sofort nach der ersten oder zweiten Stufe beendet, falls der Umbruch den internen Vorgaben entspricht. Auf den sehr aufwändigen Algorithmus der einzelnen Stufen soll hier nicht weiter eingegangen werden;

es wird auf die ausführliche Beschreibung in Römer und Voß (2010) und Voß (2007) und insbesondere Knuth und Plass (1981) verwiesen. Hier sollte nur gezeigt werden, welchen Aufwand T_EX im Gegensatz zu herkömmlichen Textverarbeitungsprogrammen treibt, um einen optimierten Umbruch zu erreichen.

Was man jetzt schon wissen sollte ...

Bei der Texterstellung mit L^AT_EX kann etwas auftreten, was bei keiner Textverarbeitung passiert und bei Anfängern zu Irritationen führen kann, weil das Prinzip oft zuerst nicht verstanden wird. Zur besseren Erklärung wird für die folgende Demonstration die Trennmöglichkeit abgeschaltet (`langwohyphens` → language without hyphens). Das Beispiel zeigt einen Text mit relativ schmaler Textbreite und den für die deutsche Sprache typischen langen Wörtern.

```
\newlanguage\langwohyphens
\AddToHook{begindocument}{\language\langwohyphens} % Trennungen jetzt nicht möglich
```

Der Begriff Transzendentalphilosophie umfasst philosophische Systeme und Ansätze, welche Grundstrukturen des allgemeinen Seins nicht durch eine Ontologie (Theorie des Seienden), sondern im Rahmen des Entstehens und Begründens von Wissen über das Sein beschreiben.

01-01-1

Der Begriff Transzendentalphilosophie umfasst philosophische Systeme und Ansätze, welche Grundstrukturen des allgemeinen Seins nicht durch eine Ontologie (Theorie des Seienden), sondern im Rahmen des Entstehens und Begründens von Wissen über das Sein beschreiben.

Wie zu sehen ist, schreibt LuaT_EX die erste und letzte Zeile über den Rand! Es bleibt natürlich die Frage, wieso kann das jede Textverarbeitung vermeiden, aber LuaT_EX nicht. Im Gegensatz zu einer Textverarbeitung, die Zwischenräume in ihrer Größe nicht begrenzt, erlaubt LuaT_EX per *Definition* nur einen maximalen Wortzwischenraum von 3.75 pt, was gerade einmal 1,4 mm entspricht. Reicht dieser Wert nicht aus, um eine Zeile vernünftig zu umbrechen, so wird automatisch das letzte Wort über den Rand hinaus geschrieben. Dieser Fall ist unter anderem sprachenabhängig; die allgemeine Wortlänge und Trennungsmöglichkeiten der zugrunde liegenden Sprache spielen eine große Rolle.

Man sollte diesen Effekt nicht als Makel, sondern als Hinweis darauf auffassen, dass der vorgegebene Text keinen Umbruch bei den strengen Vorgaben erlaubt. Er wird ohnehin in Texten mit »normaler« Textbreite und Trennmöglichkeiten selten zu beobachten sein. Kann aber bei schmalen Textspalten, beispielsweise in Tabellen, durchaus häufiger auftreten, wenn die zugrundeliegende Sprache relativ lange Silben aufweist.

Besteht keine Möglichkeit den Text zu modifizieren, so setzt man den entsprechenden Absatz »schlampig«. Dazu kann man die Umgebung `sloppypar` oder den Schalter `\sloppy` verwenden. Theoretisch könnte man den Befehl `\sloppy` bereits in der Präambel verwenden, was jedoch nicht zu empfehlen ist. Der Umbruch wird dadurch qualitativ schlechter; die Wortzwischenräume sind dann allgemein größer. Der Vollständigkeit halber sei erwähnt, dass `\fussy` die Umkehrung von `\sloppy` ist, also der Standard. Obiges Beispiel mit der Umgebung `sloppypar` gesetzt, ergibt ein besseres Erscheinungsbild:

```
\newlanguage\langwohyphens
\AddToHook{begindocument}{\language\langwohyphens} % Trennung nicht mehr möglich

\begin{sloppypar}
Der Begriff Transzendentalphilosophie umfasst philosophische Systeme und Ansätze, welche
Grundstrukturen des allgemeinen Seins nicht durch eine Ontologie (Theorie des Seienden),
sondern im Rahmen des Entstehens und Begründens von Wissen über das Sein beschreiben.
\end{sloppypar}
```

01-01-2

Der Begriff Transzendentalphilosophie umfasst philosophische Systeme und Ansätze, welche Grundstrukturen des allgemeinen Seins nicht durch eine Ontologie (Theorie des Seienden), sondern im Rahmen des Entstehens und Begründens von Wissen über das Sein beschreiben.

Grundsätzlich kann man davon ausgehen, dass die Anwendung von `sloppypar` oder `\sloppy` eher selten ist und meistens nur für die folgenden Fälle notwendig sein wird:

- Sehr schmale Textspalten, die einen Umbruch grundsätzlich erschweren.
- Längere Wörter, die aus bestimmten Gründen nicht getrennt werden sollen, beispielsweise Funktionsnamen von Programmen.
- Ungünstige URLs, die nach Vorgabe im Allgemeinen nur bei den Sonderzeichen umbrochen werden.
- Einsatz falscher Trennmuster, weil beispielsweise vergessen wurde, für ein fremdsprachiges Textzitat auch in diese Fremdsprache zu wechseln:
`\foreignquote{Sprache}{Text}` (siehe dazu Abschnitt 5.4 auf Seite 62).



Um ein Gefühl für die Häufigkeit zu bekommen, sei darauf hingewiesen, dass für dieses Buch die Anwendung der Umgebung `sloppypar` nur für den Absatz mit langen nicht-trennbaren Makronamen nötig war (siehe auf Seite 164). Festzustellen ist zudem, dass obiges Beispiel völlig unproblematisch ist, wenn die Trennungen erlaubt sind, was standardmäßig ohnehin der Fall wäre. Allerdings ist der Absatz aus \TeX -Sicht typografisch ungünstig, denn eine Trennung in der vorletzten Zeile wird grundsätzlich als schlecht bewertet (siehe Beispiel 01-01-3). Eine Alternative wäre aber nur das Erweitern der Wortzwischenräume, was eine noch schlechtere Wertung hätte.

```
\usepackage[ngerman]{babel}
```

Der Begriff Transzendentalphilosophie umfasst philosophische Systeme und Ansätze, welche Grundstrukturen des allgemeinen Seins nicht durch eine Ontologie (Theorie des Seienden), sondern im Rahmen des Entstehens und Begründens von Wissen über das Sein beschreiben.

01-01-3

Der Begriff Transzendentalphilosophie umfasst philosophische Systeme und Ansätze, welche Grundstrukturen des allgemeinen Seins nicht durch eine Ontologie (Theorie des Seienden), sondern im Rahmen des Entstehens und Begründens von Wissen über das Sein beschreiben.

Mit den Methoden der Mikrotypografie durch das Laden des Paketes `microtype` von Robert Schlicht kann man das Layout verbessern:

```
\usepackage{microtype}
\usepackage[ngerman]{babel}
```

Der Begriff Transzendentalphilosophie umfasst philosophische Systeme und Ansätze, welche Grundstrukturen des allgemeinen Seins nicht durch eine Ontologie (Theorie des Seienden), sondern im Rahmen des Entstehens und Begründens von Wissen über das Sein beschreiben.

01-01-4

Der Begriff Transzendentalphilosophie umfasst philosophische Systeme und Ansätze, welche Grundstrukturen des allgemeinen Seins nicht durch eine Ontologie (Theorie des Seienden), sondern im Rahmen des Entstehens und Begründens von Wissen über das Sein beschreiben.

1.2 Warum Lua^AT_EX und nicht pdf^AL^AT_EX?

Man kann zur Zeit noch davon ausgehen, dass der größere Teil der Nutzer von L^AT_EX seine Dokumente mit pdf^AL^AT_EX erstellen dürfte. Das Hauptargument dafür ist schlicht und ergreifend »never change a running system«. Mit der Anwendung von pdf^AL^AT_EX benutzt man faktisch ein abgekapseltes System, welches Quelldokumente verarbeitet, die völlig unabhängig vom Betriebssystem und der zugrundeliegenden T_EX-Distribution sind. Dieser Vorteil ist auch gleichzeitig ein Nachteil: Sobald man in seinem Dokument sprachenspezifische Dinge, wie Kodierung und Sortierung von fremdsprachlichen Zeichenketten beachten muss, steigt der Aufwand mit pdf^AL^AT_EX teilweise erheblich. Mit Lua^AT_EX, was eine vollwertige Unicode-Unterstützung ermöglicht, muss darauf faktisch keine Rücksicht genommen werden. Hat man eine Schrift, die die entsprechenden Zeichen enthält, können fremdsprachliche Textzitate eingefügt werden, ohne dass man irgendwelche speziellen Pakete laden muss. Auch die Änderung der Schreibrichtung ist möglich, wenn eine der RTL-Sprachen (Right-To-Left) verwendet wird.

。たしまきでん学はに時、に前年の本日、ていでん住にシリルベは私
 Ωστόσο, ποτέ δεν έμαθαν ελληνικά και γνωρίζω περισσότερα
 γράμματα μόνο και μόνο επειδή είχα ανάγκη αυτό για τα μαθηματικά.
 داي إيزيچ فوراستسيونغ إست، داس مان این سكریفت بینوتست، داي أوتش دییز زیتشن دارستیلن کان. فی
 دیسم تقع ناتورلیش این آرایبش سكریفت.

Durch die Verwendung von Schriften im Format TrueType oder OpenType ist dies im Allgemeinen kein Problem mehr. So wurde in obigem Beispiel temporär für Japanisch auf die Schrift Kozuka Gothic Pro², für Griechisch auf Lucida Sans Unicode³ und für Arabisch auf die TrueType-Schrift ArabicTypesetting⁴ umgeschaltet. Die Definitionen lauten:

```
\newfontfamily\LucidaUnicode[Scale=0.9]{LucidaSansUnicode}
\newfontfamily\KozGo[Scale=0.88]{KozGoPro-Regular}
\newfontfamily\Arabic[Scale=MatchUppercase,Script={Arabic}]{ArabicTypesetting}
```

²<http://www.adobe.com/products/type/font-designers/masahiko-kozuka.html>

³https://en.wikipedia.org/wiki/Lucida_Sans_Unicode

⁴<https://www.microsoft.com/typography/fonts/font.aspx?FMID=1704>

1.2.1 LuaTeX oder Lua^ATeX?

Im Umgang mit $\text{L}^{\text{A}}\text{TeX}$ wird der Anfänger mit einer Vielzahl an Programmnamen konfrontiert, deren Einordnung nicht einfach ist. Auch geht es begrifflich ziemlich durcheinander: Es wird immer vom Programm $\text{L}^{\text{A}}\text{TeX}$, beziehungsweise $\text{LuaL}^{\text{A}}\text{TeX}$ gesprochen, obwohl es im eigentlichen Sinne keine eigenständigen Programme sind. Das zugrundeliegende Programm ist meistens $\text{T}_{\text{E}}\text{X}$, $\text{pdfT}_{\text{E}}\text{X}$ oder $\text{LuaHbT}_{\text{E}}\text{X}$, wobei es noch weitere Varianten gibt. Allen ist gemeinsam, dass sie auf dem originalen $\text{T}_{\text{E}}\text{X}$ von Donald Knuth, einschließlich der Erweiterung durch $\text{eT}_{\text{E}}\text{X}$, basieren, welches mit jeder $\text{T}_{\text{E}}\text{X}$ -Distribution zur Verfügung steht. Das von Taco Hoekwater und anderen entwickelte $\text{LuaHbT}_{\text{E}}\text{X}$ ist eine Weiterentwicklung von $\text{T}_{\text{E}}\text{X}$, welches die vielen Beschränkungen von $\text{T}_{\text{E}}\text{X}$ aufhebt und zudem die Integration der Skriptsprache Lua in $\text{T}_{\text{E}}\text{X}$ -Dokumenten erlaubt.

This is LuaHBTeX, Version 1.14.0 (TeX Live 2021)

Execute 'luahbtex --credits' for credits and version details.

There is NO warranty. Redistribution of this software is covered by the terms of the GNU General Public License, version 2 or (at your option) any later version. For more information about these matters, see the file named COPYING and the LuaTeX source.

LuaTeX is Copyright 2021 Taco Hoekwater and the LuaTeX Team.

$\text{L}^{\text{A}}\text{TeX}$ wiederum ist nichts anderes als eine Sammlung (Bibliothek) von Makros (Befehlen), die in die jeweilige $\text{T}_{\text{E}}\text{X}$ -Variante fest eingebunden sind. Unter Linux ist $\text{LuaL}^{\text{A}}\text{TeX}$ daher nichts anderes als ein Link (Verweis) auf $\text{LuaHbT}_{\text{E}}\text{X}$. Da Windows dieses Verlinken nicht unterstützt, ist hier $\text{LuaL}^{\text{A}}\text{TeX}$ im Prinzip eine Kopie von $\text{LuaHbT}_{\text{E}}\text{X}$ mit eingebundenen $\text{L}^{\text{A}}\text{TeX}$ -Makros. Hierbei steht »hb« für HarfBuzz⁵ (<https://de.wikipedia.org/wiki/HarfBuzz>⁵ (<https://harfbuzz.github.io/>), welches eine freie Grafikbibliothek für Schriften ausschließlich im Format OpenType ist und nun standardmäßig in $\text{LuaT}_{\text{E}}\text{X}$ eingebunden ist. Unterstützt werden vorrangig indische, orientalische und Thai-Sprachen. Aktiviert werden muss die HarfBuzz-Engine über das optionale Argument beim Laden einer Schrift, was im folgenden Beispiel für Burmesisch und Arabisch gezeigt wird. Im ersten Fall ist das Wort jeweils ohne HarfBuzz und im zweiten Fall dann mit HarfBuzz gesetzt worden. Man erkennt deutlich die Anwendung der Ligaturen bei Benutzung von HarfBuzz.



```
\usepackage{fontspec}
\newfontfamily\Arab{ArabicTypesetting}[Scale=3, Renderer=Basic, Script={Arabic}]
\newfontfamily\ArabHB{ArabicTypesetting}[Scale=3, Renderer=Harfbuzz, Script={Arabic}]
\newfontfamily\Burmese{NotoSerifMyanmar-Regular.ttf}[Renderer=Basic,Script=Myanmar,Scale=1.5]
\newfontfamily\BurmeseHB{NotoSerifMyanmar-Regular.ttf}[Renderer=Harfbuzz,Script=Myanmar,Scale=1.5]
\newenvironment{RTL}% für Arabisch
{
\par \pardir TRT\textdir TRT\bodydir TRT
}{\par}

\Arab\begin{RTL}ديضنتللا\end{RTL}

\ArabHB\begin{RTL}ديضنتللا\end{RTL}

\Burmeseမုတီခင်းလောင်းကိ
\BurmeseHBမုတီခင်းလောင်းကိ
```

⁵Persisch für OpenType

01-02-1

التنضيد العربي
التنضيد العربي

မူပိုင်ခွင့်လက်မှတ်
မူပိုင်ခွင့်လက်မှတ်

1.2.2 Die TeX-Distribution

Das TeX-System als Ganzes ist eine ziemlich komplexe Angelegenheit, sodass man für eine Installation auf eine sogenannte Distribution zurückgreift, die sich »um alles kümmert«. Es gibt im Prinzip nur zwei freie Distributionen: TeXLive⁶ und MiKTeX⁷. TeXLive unterstützt ebenso wie MiKTeX alle bekannten Betriebssysteme, wie Windows, macOS, Linux in fast allen Varianten. Für welche der beiden man sich entscheidet, ist unerheblich; das Ergebnis, die erzeugte PDF-Datei wird identisch sein. Daneben gibt es noch einige kommerzielle Varianten, auf die hier jedoch nicht weiter eingegangen wird.

1.3 Das Prinzip

Es wurde bereits darauf hingewiesen, dass man sich bei der Erstellung des Textes eher als *Programmierer* denn als *Texteingabe* fühlen sollte. Dies erleichtert das Verständnis für das Prinzip bei der Texteingabe. Bei einer optimalen Situation kann man bei Auswahl einer bestimmten Dokumentenklasse erreichen, dass man lediglich die Sprachauswahl, Eingabekodierung und Fontkodierung vorgeben muss. Sämtliche anderen Vorgaben, die das Layout des Dokumentes betreffen, sind bereits in der gewählten Dokumentenklasse vorgegeben, sodass der sogenannte Übersetzungsvorgang ein fertig formatiertes Dokument höchster Qualität liefert.

Abbildung 1.5 auf der nächsten Seite zeigt den Ablauf eines Übersetzungsvorganges zur Erzeugung eines Dokumentes, wie es der augenblickliche Standard ist. Die dicke rote Linie kennzeichnet den in dieser Veröffentlichung beschriebenen Ablauf, der ebenso auch für dieses Buch selbst der Standard war. Der Quelltext muss wie jeder andere Quelltext eines beliebigen Programmes, beispielsweise C++ oder Perl, syntaktisch korrekt sein, wenn eine fehlerfreie Ausgabe erfolgen soll. Je nach der Komplexität des Dokumentes kann es notwendig sein, mehrere Durchläufe mit dem TeX-Compiler und anderer externer Programme vorzunehmen. Zu diesen externen Programmen zählen insbesondere solche zur Erstellung einer Bibliografie, eines Indexes oder Glossars. Die Anforderungen an den Anwender sind hierbei gering, selbst für den Fall, dass er nicht mit einem Makefile oder einer grafischen Entwicklungsumgebung (GUI) für L^AT_EX arbeitet. Die gesamte

⁶<http://www.tug.org/texlive>

⁷<http://www.miktex.org>

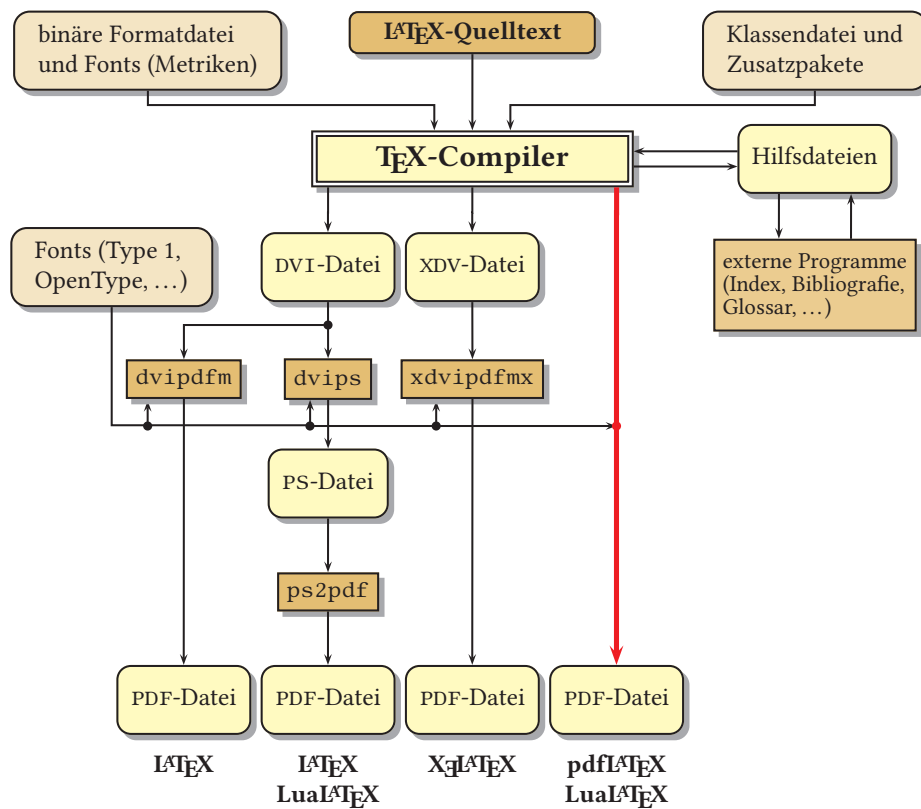


Abbildung 1.5: »Von der Quelle bis zur Mündung ...« Der Aufbau eines T_EX-Systems mit pdfT_EX, X_eT_EX oder LuaT_EX als Compiler, die eine Ausgabe im PDF-Format erlauben.

Schrifteinbindung, die in der Abbildung 1.5 dargestellt ist, erfolgt vollautomatisch und muss den Anwender nur in seltenen Fällen interessieren.

1.4 Externe L^AT_EX-Anwendungen

Mit LuaL^AT_EX wird das Dokument erstellt, welches den reinen Text mit allen Querverweisen enthält. Ohne weitere externe Programme können aber zusätzliche Elemente, wie Bibliografie, Index, Glossar, usw. im Allgemeinen nicht erstellt werden. Die Tabelle 1.1 zeigt eine Zusammenstellung einiger Programme, die automatisch bei der Installation einer T_EX-Distribution zur Verfügung stehen.

Tabelle 1.1: Auswahl einiger Programme, die mit jeder T_EX-Distribution installiert werden.

Name	Bedeutung
tex	Das Original ...
latex	T _E X mit dem Format L ^A T _E X.
luatex	Das »neueste« T _E X

<i>Name</i>	<i>Bedeutung</i>
luahbtex	Das »neueste« T _E X mit Unterstützung für HarfBuzz
lualatex	LuahbT _E X mit dem Format L ^A T _E X.
dviluatex	LuahbT _E X mit einer Ausgabe im DVI-Format.
dvilualatex	Lual ^A T _E X mit einer Ausgabe im DVI-Format.
biber	Das Programm zum Verarbeiten von Bibliografieverweisen im Zusammenhang mit einer B _I B _T E _X -kompatiblen Datenbank.
bibtex	Veraltetes Programm zur Erzeugung einer Bibliografie.
makeindex	Ein Programm zum Verarbeiten von Indexeinträgen.
xindex	Ein neueres unicodefähiges Programm zum Verarbeiten von Indexeinträgen.
xindy	Ein Programm zum Verarbeiten von Indexeinträgen.
makeglossaries	Das Programm zum Verarbeiten von Glossareinträgen.
texcount	Zählen der Wörter eines T _E X-Dokumentes.
texdoc	Anzeigen von Paket- und Dokumentenklassendokumentationen.
tex4ebook	Konvertieren eines L ^A T _E X-Dokumentes in ein EBook-Format.
pdfcrop	»Croppen« (Entfernen der weißen Ränder) eines PDF-Dokuments.
pdfbook	Drucken eines PDF-Dokuments im Buchformat (zweiseitig).
texhash	Aktualisieren der Datei-Datenbank eines T _E X-Systems (File Name Data Base).
arara	Skript zum automatisierten Verarbeiten eines L ^A T _E X-Dokuments.
latexmk	Skript zum automatisierten Verarbeiten eines L ^A T _E X-Dokuments.
exeltex	Skript zum Exportieren von Excel-Tabellen nach L ^A T _E X.
asy	Skript zum Ausführen von Quellcode für das Grafiksystem Asymptote.
kpsewhich	Bestimmung des Speicherortes (Verzeichnisses) einer Datei, die Teil des T _E X-Systems ist.

