

# Notizen, Hinweise und Links zu TeX

Thomas Klampfl

31.12.2024

## Bücher:

LaTeX: A Document Preparation System, Leslie Lamport, Addison Wesley, 2nd ed, 1994.

The LaTeX Companion, Mittelbach, Goossens with Braams, Carlisle and Rowley, Addison Wesley, 2nd ed, 2004. 3rd ed. 2023.

The LaTeX Graphics Companion, Goossens, Mittelbach and Rahtz, Addison Wesley, 1997.

Books about TeX, typography, and friends: <https://tug.org/books/>

## Tutorials:

Learn LaTeX: <https://www.learnlatex.org>

## Dokumentation:

Tobias Oetiker, Hubert Partl, Irene Hyna and Elisabeth Schlegl, The Not So Short Introduction to LaTeX 2 $\epsilon$ : <https://ctan.org/pkg/lshort-english>

LaTeX2 $\epsilon$ : An unofficial reference manual: <https://tug.org/texinfohtml/latex2e.html>

## Distributionen:

TeX Live: <https://tug.org/texlive/>

The MacTeX: <https://tug.org/mactex/>

MiKTeX: <https://miktex.org/>

**Comprehensive TeX Archive Network:** <http://www.ctan.org>

**The LaTeX Project:** <https://www.latex-project.org/>

## Engines:

TeX: Von Donald E. Knuth zwischen 1977 und 1986 entwickeltes Textsatzsystem für hochwertigen Textsatz insbesondere von mathematischen Formeln. Es werden dvi Dokumente erzeugt, die mit Zusatzprogrammen nach PostScript (dvips - <https://tug.org/dvips/>) oder pdf (dvipdfmx - <https://tug.org/dvipdfmx/>) transformiert werden können. Bei TeX handelt es sich um einen Interpreter, der circa 300 Primitive kennt, und einen Mechanismus zur Definition von Makros umfasst. LaTeX ist ein Makropaket, das von TeX geladen wird, und höhergradige Anweisungen für den Textsatz bietet. TeX war in der Programmiersprache WEB basierend auf Pascal geschrieben worden, und bildet in einer Portierung nach C (Web2C -

<https://tug.org/web2c/>) die Grundlage neuerer TeX-Distributionen.

Für TeX-Systeme wurde eine eigene Verzeichnisstruktur definiert: Standard TeX Directory Structure (TDS – <https://tug.org/tds/>). Ein Programm namens Kpathsea (<https://tug.org/kpathsea/>) ist integriert, das Dateien bzw. Pakete zu finden erlaubt. Es wird eine Datenbank (ls-R) unterhalten, die von Kpathsea durchsucht wird; wenn die Datei nicht gefunden werden konnte, wird das Dateisystem durchsucht.

TeX arbeitet nur mit den Abmessungen von Glyphen; die Schriftzeichen werden erst sekundär eingesetzt. Ursprünglich waren die Schriftsätze in MetaFont (<https://ctan.org/pkg/metafont>) definiert. Dieses wurde später durch MetaPost (<https://tug.org/metapost.html>) ersetzt. MetaPost ist ein mächtiges Werkzeug zur Erzeugung von Graphiken in skalierbarem PostScript.

$\varepsilon$ -TeX: Nimmt einige Erweiterungen von TeX vor; z.B. die wesentliche Erhöhung der Anzahl der verfügbaren Register oder Unterstützung für Text, der von Rechts nach Links läuft (`\beginL`, `\endL`, `\beginR`, und `\endR`). All diese Veränderungen sind in pdfTeX eingegangen, sodass  $\varepsilon$ -TeX nicht direkt benutzt wird. (<https://www.ctan.org/pkg/etex>)

pdfTeX: Liefert als Ergebnis direkt pdf Dateien und unterstützt standardmäßig Mikrotypographie. Es kann mit Type 1 und TrueType Schriftsätzen umgehen; diese müssen allerdings konfiguriert werden: TeX arbeitet nur mit sogenannten tfm Dateien, die im wesentlichen die Abmessungen der Zeichen beinhalten. Diese Dateien tragen einen Namen, der für die Selektion der Schrift in TeX verwendet wird. Um ein graphisches Ergebnis zu erhalten, muss eine Assoziation hergestellt werden zwischen dem Namen der TeX Font Metrics Datei und der Fontdatei. Dies geschieht in sogenannten map Dateien, die sowohl von dvi Viewern und Transformatoren ausgelesen werden, als auch von pdfTeX. Obwohl in LaTeX mit pdfTeX mit Unicode Kodierung gearbeitet werden kann (seit 2018 default Einstellung), funktioniert pdfTeX intern mit ASCII / 8 Bit. Die Eingabe eines TeX Prozesses ist eine Datei, die Octets beinhaltet, die Zeichen repräsentieren. Um diese korrekt zu interpretieren bedarf es eines Mappings dieser Octets zu abstrakten Zeichen. TeX nimmt nun an, dass Zeichen aus dem Bereich des sichtbaren ASCII (dezimal 32-126) die Nummern haben, sowie sie in der ASCII Code Tabelle definiert sind. Alle anderen Zeichen werden durch LaTeX internal character representation Objekte vertreten. So ist die interne Repräsentation des scharfen s („ß“) das LICR Objekt `\ss`. Um nun zu bestimmen, wie der Eingabestrom zu den internen Objekten gemappt wird, wird ein *Input Encoding* definiert, d.h. es wird ein Parameter an das Paket `inputenc` übergeben (<https://www.ctan.org/pkg/inputenc>). Auf Ausgabeseite gibt es ein Mapping zwischen den internen Repräsentationen und den Slots in einem bestimmten Zeichensatz. Es wird wiederum das Encodingschema als Parameter an das Paket `fontenc` übergeben. (<https://www.ctan.org/pkg/fontenc>) – Das ganze Thema wird dann relevant, wenn plötzlich Zeichen im pdf Dokument fehlen (meist Symbole), oder unterschiedliche Sprachen bzw. Schriftsysteme miteinander kombiniert werden sollen. - Zur Unterstützung verschiedener Sprachen in LaTeX / pdfTeX gibt es das Paket `babel`, dem die benutzten Sprachen als Parameter zu übergeben sind. (<https://www.ctan.org/pkg/babel>) Längere Zeit benutzte man `babel` für pdfTeX und für XeTeX das Paket `polyglossia` (<https://www.ctan.org/pkg/polyglossia>); mittlerweile unterstützt `babel` auch XeTeX und LuaTeX. (<https://tug.org/applications/pdftex/index.html>)

Omega / Aleph: Omega ist eine TeX Erweiterung, die auch intern Unicode verwendet. Es wurde entwickelt, um möglichst alle Schriftsysteme mittels eines Systems verarbeiten zu können. Dafür wurden neue Schreibrichtungsdirektiven eingeführt: Es muss möglich sein, die Schreibrichtung der Seite zu definieren (primäre Schreibrichtung), die Schreibrichtung der Zeile (sekundäre Schreibrichtung) und die Schreibrichtung des einzelnen Zeichens (tertiäre Schreibrichtung). Es gibt dementsprechend vier Möglichkeiten für die Position des ersten Zeichens auf der Seite, die Position

des ersten Zeichens in der Zeile, und die Orientierung eines einzelnen Zeichens: T (= Top), R (= Right), B (= Bottom), L (= Left). Gesetzt werden können diese Richtungen für die Parameter: `\pagedir` Die Richtung der Seite; `\bodydir` Die Richtung von vboxes (= vertical boxes); dies beinhaltet den Haupttextfluß; `\pardir` Die Richtung für Paragraphen; `\textdir` Die Richtung des Textes und von hboxes (= horizontal boxes); `\mathdir` Die Richtung für mathematischen Textsatz. Omega führt auch das Feature des sogenannten *Omega Translation Process* ein. Damit ist es möglich einen Eingabestrom regelbasiert in einen Ausgabestrom zu übersetzen. Ein einfaches Beispiel ist etwa das griechische Sigma, das in Endposition eine Finalform annimmt. (U+03C3: σ, U+03C2: Ϻ) Diese besitzt einen eigenen Codepoint, ist aber nur eine Präsentationsform, und sollte deshalb z.B. in einer Suche keinen Unterschied machen, weshalb sie nur in der Darstellung unterschieden werden sollte. Omega wurde durch Aleph ersetzt. Es wurde nur der bidirektionale Textsatz gründlicher getestet (vgl. z.B. Makor 2 - Typeset pointed Hebrew using Omega, <https://www.ctan.org/pkg/makor2>), die Konzepte wurden aber nach LuaTeX übernommen.

(<https://www.ctan.org/pkg/omega>, <https://www.ctan.org/pkg/aleph>)

**XeTeX:** XeTeX ist eine TeX Variante, die Unicode (UTF-16) als internes Repräsentationsformat benutzt, und moderne Fonttechnologien (OpenType Fonts, TrueType Fonts, Apple Advanced Typography [AAT], Graphite Fonts) unterstützt. (<https://www.ctan.org/pkg/xetex>) Traditionell wurde für XeTeX das Paket `polyglossia` für die Sprachauswahl verwendet. In einer bestimmten Sprache zu schreiben, bedeutet für TeX eine Reihe von Änderungen vorzunehmen: Laden der Muster für das Abteilen von Wörtern, Umschalten des Schriftsatzes, wenn dies spezifiziert wurde, Auswahl von Fontfeatures für eine bestimmte Sprache, Anpassen einiger typographischer Konventionen, Übersetzen von festgelegten Strings, wie „Chapter“, „Table of Contents“, usw, Umstellen des Datumssystems, Umstellen der Nummerierung, Wahl der Schreibrichtung. (<https://www.ctan.org/pkg/polyglossia>) Um z.B. ein Dokument in Deutsch zu schreiben, das auch griechischen Text beinhaltet, kann folgendes Beispiel dienen:

## 1 Einleitung

Ein deutscher Text mit eingebettetem griechischen Text λέγει in eigener Schriftart.

ὥσπερ' τῶν αἰσθητῶν αἰγυπτίων σώζων τοὺς ἐξ' ἰσραὴλ' ἄρματα Φαραῶ καὶ τὴν δύναμιν αὐτοῦ ἔρριπεν εἰς θάλασσαν· οὕτω καὶ πάντων ἀνθρώπων σώζων τὸ γένος, τῶν νοητῶν αἰγυπτίων. συνέτριψεν τὰ ἄρματα·

```
\documentclass[11pt,a4paper]{article}
\usepackage{polyglossia}
\setmainlanguage[variant=austrian,spelling=new]{german}
\setotherlanguage[variant=polutonic]{greek}
\newfontfamily\greekfont[Script=Greek,ExternalLocation="./greek-fonts/"]{SBL_BLit.ttf}
\begin{document}
\section{Einleitung}
```

Ein deutscher Text mit eingebettetem griechischen Text `\foreignlanguage{greek}{λέγει}` in eigener Schriftart. \par

```
\begin{greek}
ὥσπερ' τῶν αἰσθητῶν αἰγυπτίων σώζων τοὺς ἐξ' ἰσραὴλ' ἄρματα Φαραῶ καὶ τὴν δύναμιν
αὐτοῦ ἔρριπεν εἰς θάλασσαν· οὕτω καὶ πάντων ἀνθρώπων σώζων τὸ γένος, τῶν
νοητῶν αἰγυπτίων. συνέτριψεν τὰ ἄρματα·
\end{greek}
\end{document}
```

LuaTeX: LuaTeX ist eine Weiterentwicklung von pdfTeX, und soll die Standardengine für TeX Textsatz werden. Es kann Schriftarten in TrueType, OpenType und Type-1 handhaben. Es wurde, neben anderen Features, das Schreibrichtungsmodell von Omega übernommen, aus dem Schriftsatzdesignprogramm FontForge (<https://fontforge.org/>) Bibliotheken zum Umgang mit Schriften, die MetaPost Bibliothek wurde eingebunden, und schließlich die Skriptsprache Lua (<https://www.lua.org/>) integriert. Mittels LuaTeX kann Lua-Code in TeX ausgeführt werden, und umgekehrt. Alle Interna von TeX können abgefragt und verändert werden. Das default Encoding ist UTF-8. (<https://www.ctan.org/pkg/luatex>; <http://www.luatex.org/>)

Beispiel für das Einbinden von TrueType Schriftsätzen mit LuaTeX:

```
\documentclass[11pt,a4paper]{article}
\usepackage[polutonikogreek,ngerman]{babel}
\usepackage{fontspec}
\setmainfont{GentiumBookPlus-Regular}[
  Extension = .ttf ,
  Path = ./gentium-sil/ ,
  BoldFont = GentiumBookPlus-Bold ,
  ItalicFont = GentiumBookPlus-Italic ,
  BoldItalicFont = GentiumBookPlus-BoldItalic ]
\parindent0pt
\begin{document}
\section{Test für Gentium Fonts}
Normaler Text \textbf{fetter Text} \textit{italic Text} \textit{\textbf{fett-italic Text}}.\par
\vspace{3mm}
\selectlanguage{greek}
ὥσπερ' τῶν αἰσθητῶν αἰγυπτίων σώζων τοὺς ἐξ' ἰσραὴλ· ἄρματα Φαραῶ καὶ τὴν δύναμιν
αὐτοῦ ἔρριψεν εἰς θάλασσαν· οὕτω καὶ πάντων ἀνθρώπων σώζων τὸ γένος, τῶν
νοητῶν αἰγυπτίων. συνέτριψεν τὰ ἄρματα.\par
\vspace{3mm}
\selectlanguage{ngerman}
Weiterer Text.
\end{document}
```

## 1 Test für Gentium Fonts

Normaler Text **fetter Text** *italic Text* ***fett-italic Text***.

ὥσπερ' τῶν αἰσθητῶν αἰγυπτίων σώζων τοὺς ἐξ' ἰσραὴλ· ἄρματα Φαραῶ καὶ τὴν  
 δύναμιν αὐτοῦ ἔρριψεν εἰς θάλασσαν· οὕτω καὶ πάντων ἀνθρώπων σώζων τὸ  
 γένος, τῶν νοητῶν αἰγυπτίων. συνέτριψεν τὰ ἄρματα·

Weiterer Text.

### Graphik:

asymptote – 2D and 3D TeX-Aware Vector Graphics Language  
 (<https://www.ctan.org/pkg/asymptote>)

metapost – A development of METAFONT for creating graphics (PostScript, SVG)

(<https://www.ctan.org/pkg/metapost>)

pgf / TikZ— Create PostScript and PDF graphics in TeX (<https://www.ctan.org/pkg/pgf>)

### Präsentationen / Slides:

beamer - A LaTeX class for producing presentations and slides  
(<https://www.ctan.org/pkg/beamer>)

### Alternative Dokumentklassen:

koma-script - A bundle of versatile classes and packages  
(<https://www.ctan.org/pkg/koma-script>)

memoir - Typeset fiction, non-fiction and mathematical books  
(<https://www.ctan.org/pkg/memoir>)

### Beispiel: Kritische Edition mit Paralleltexten

#### ΨΑΛΜΟΣ Α'

#### Psalm 1

##### Expositio 1: Hypothesis

- 1 Τὴν ἀρχὴν τῆς προφητείας τῷ ἐξ αὐ-  
τοῦ κατὰ σάρκα τεχνησομένῳ Χριστῷ,  
3 ἀνατίθῃσιν ὁ Δαυῖδ· διὸ ἐν πρώτοις μα-  
καρίζει τοὺς εἰς αὐτὸν ἡλπικότας· μα-  
5 καρίους δὲ καλεῖ, τοὺς μὴ πορευθέν-  
τας ἐν βουλῇ ἀσεβῶν· μήτε μὴν στάν-  
7 τας ἐν ὁδῷ ἁμαρτωλῶν· μήτε ἐπὶ κα-  
θέδραν λοιμῶν καθεστηκότας· τρία γὰρ  
9 ἦν τάγματα παρὰ Ἰουδαίοις κατὰ τοῦ  
σωτῆρος ἐπαναστάντα· γραμματεῖς· φα-  
11 ρισαῖοι· καὶ νομικοί· οἱ καὶ κληθεῖεν  
εἰκότως, ἀσεβεῖς· καὶ ἁμαρτωλοὶ· καὶ  
13 λοιμοί· —

Den Anfang der Prophetie bringt Da-  
vid Christo dar, der dem Fleisch nach  
aus ihm geboren werden soll. Des-  
halb preist er zuerst die selig, die auf  
ihn hoffen. Er nennt selig jene, die  
weder im Rat der Gottlosen wandel-  
ten, noch auf den Weg der Sünder  
traten und sich auch nicht auf den  
Stuhl der Verseuchten setzten. Denn  
es gab bei den Juden drei Gruppen,  
die sich gegen den Erlöser erhoben:  
Schriftgelehrte, Pharisäer und Geset-  
zeskundige. Eben diese könnten zu  
Recht Gottlose, Sünder und Verseuch-  
te genannt werden.

*txt* V1 M O B1 B2 V5 P7

κατὰ σάρκα τεχνησομένῳ Χριστῷ ἀνατίθῃσιν ] τεχνησο[μένῳ] [ἀνατίθῃσιν(?)] B2  
— ὁ Δαυῖδ ] ὁ μέγας Δαυῖδ V5 P7 — ἡλπικότας· μακαρίους δὲ καλεῖ ] post ἡ- non  
legi possunt B2 — μὴ πορευθέντας ] μήτε πορ[ε]υ[θέντ]ας B2 — μὴν ] -ν supra lin.  
O — ἐν ὁδῷ ] non legi potest B2 — μήτε ἐπὶ καθέδραν ] μὴ δὲ ἐπὶ καθέδρας V5 P7  
— ἐπὶ καθέδραν ] ἐπὶ καθέδρα O — λοιμῶν καθεστηκότας ] κεκαθικώτας λοιμῶν B1

Aufbau der Quelldatei:

```

div @type = psalm @n = 1
  div @type = psalmverse @n = 1.0
    div @type = text @xml:lang = grc
      div @type = psalmtext
        quote @type = psalmtext @xml:id @subtype = heading
          Psalmzitat
        div @type = commentary
          p @xml:id @n = 1 @ana = hypothesis
            Kommentarfragment
          div @type = links
            p
              listRef
                ref @target
          div @type = textcritic
            app @type = fragment
              rdg @wit
            app @type = text
              lem Lemma
              rdg @wit Lesart
          div @type = translation @xml:lang = de
            quote @corresp = id @type = psalmtext
              Übersetzung des Psalmtextes
            p @n @corresp = id
              Übersetzung des Kommentarfragments

```

Es handelt sich um die Edition eines Psalmenkommentars, d.h. die Struktur ist durch die Psalmen mit ihren Versen gegeben. Zu den Psalmversen gibt es – zumeist kurze – Kommentare, die aus mehreren Handschriften rekonstruiert werden. Zu dem Psalmtext gibt es eine Übersetzung; zu den Kommentaren ebenfalls. Die Assoziation zwischen griechischem Text und der Übersetzung erfolgt mittels des `corresp` Attributs, das auf die `@xml:id` des griechischen Textes zeigt. Zu jedem Kommentar gibt es eine sogenannte Bezeugungsleiste, d.h. eine Liste der Handschriften, die den Text bieten (`app @type = fragment`). Jede Textvariante wird mit dem Lemma und der abweichenden Lesart, sowie den Textzeugen notiert (`app @type = text`). Für die Verlinkung der Kommentarfragmente mit den Transkriptionen der einzelnen Handschriften wird eine Liste von Referenzen (`ref`) geführt.

Die Darstellung soll als Paralleledition mit Zeilennummerierung erfolgen, mit einem angehängten kritischen Apparat. Dazu werden die LaTeX Pakete `reledmac` und `reledpar` verwendet. Das Grundgerüst sieht folgendermaßen aus:

```

\begin{pairs}
  \begin{Leftside}
    \beginnumbering
    \pstart
    \firstlinenum{1}
    \linenumincrement{2}
    Text
  \pend
  \endnumbering
\end{Leftside}
\begin{Rightside}
  \beginnumbering
  \numberlinefalse

```

```

        \pstart
        Text
        \pend
        \endnumbering
    \end{Rightside}
\end{pairs}
\Columns

```

Das Template für den Psalm erzeugt ein Kapitel mit einer Überschrift:

```

<xsl:template match="tei:div[@type = 'psalm']">
    \chapter{
        <xsl:text>Psalm </xsl:text>
        <xsl:value-of select="@n" />
    }
    <xsl:apply-templates select="tei:div[@type = 'psalmverse']" />
</xsl:template>

```

Das Template für den Psalmvers ruft nur die weitere Verarbeitung des eingebetteten divs auf. Dadurch wird ausgeschlossen, dass die benachbarten divs direkt verarbeitet werden (div @type = links, div @type = translation).

```

<xsl:template match="tei:div[@type = 'psalmverse']">
    <xsl:apply-templates select="tei:div[@type = 'text']" />
</xsl:template>

```

Die Regel für den Text macht wiederum nichts, sondern ruft nur die folgenden Templates für die Verarbeitung des Psalmtextes, des Kommentars und des kritischen Apparats auf.

```

<xsl:template match="tei:div[@type = 'text']">
    <xsl:apply-templates
        select="tei:div[@type = 'psalmtext']
        | tei:div[@type = 'commentary']
        | tei:div[@type = 'textcritic']" />
    \vspace{7mm}
</xsl:template>

```

Die entsprechenden Templates tun nichts, und rufen die Templates für ihre jeweiligen Kindknoten auf.

```

<xsl:template match="tei:div[@type = 'psalmtext']">
    <xsl:apply-templates select="child::node()" />
</xsl:template>

```

```

<xsl:template match="tei:div[@type = 'commentary']">
    <xsl:apply-templates select="child::node()" />
</xsl:template>

```

```

<xsl:template match="tei:div[@type = 'textcritic']">
    <xsl:apply-templates select="child::node()" />
</xsl:template>

```

Kind des `div @type = psalmtext` ist das `quote` Element mit dem Psalmtext. Für dieses gibt es das folgende Template, das nur für griechischen Text greift. Es wird das Kommentarfragment in einer Umgebung für griechischen Text in der linken Spalte gesetzt, und dann die Übersetzung innerhalb der rechten Spalte für die weitere Verarbeitung selektiert.

```
<xsl:template match="tei:quote[(@type = 'psalmtext')
                                and ancestor::tei:div[@xml:lang = 'grc']]">
  \begin{pairs}
    \begin{Leftside}
      \beginnumbering
      \numberlinefalse
      \pstart{
        <xsl:if test="exists(@n)">
          <xsl:text>(</xsl:text>
          <xsl:value-of select="@n"/>
          <xsl:text>) </xsl:text>
        </xsl:if>
        \foreignlanguage{greek}{
          <xsl:value-of select="text()" />
        }
      }
    \pend
    \endnumbering
  \end{Leftside}
  \begin{Rightside}
    \beginnumbering
    \numberlinefalse
    \pstart{
      \foreignlanguage{german}{
<xsl:apply-templates
select="parent::tei:div/parent::tei:div/parent::tei:div/
tei:div[@type = 'translation']/
tei:quote[current()/@xml:id = substring-after(@corresp, '#')]" />
      }
    }
    \pend
    \endnumbering
  \end{Rightside}
\end{pairs}
\Columns
</xsl:template>
```

Es folgt das Template für den Text der Übersetzung des Psalmtextes.

```
<xsl:template match="tei:quote[(@type = 'psalmtext')
                                and ancestor::tei:div[@xml:lang = 'de']]">
  <xsl:if test="exists(@n)">
    <xsl:text>(</xsl:text>
    <xsl:value-of select="@n"/>
    <xsl:text>) </xsl:text>
  </xsl:if>
  <xsl:value-of select="text()" />
</xsl:template>
```



Das nächste Template verarbeitet den griechischen Kommentartext sowie die parallele Übersetzung.

```
<xsl:template
  match="tei:p[ancestor::tei:div[@xml:lang = 'grc']]
    [not(parent::tei:div[@type = 'links'])]">
  \par
  \vspace{3mm}
  \begin{german}
    <xsl:text>\textbf{Expositio }</xsl:text>
    <xsl:value-of select="@n"/>
    <xsl:text>:</xsl:text>
    <xsl:if test="exists(@ana)
      and @ana = 'hypothesis'">Hypothesis</xsl:if>
  <xsl:if
    test="exists(parent::tei:div[@type = 'commentary']/
      @subtype)">
    <xsl:text> (</xsl:text>
    <xsl:value-of
      select="parent::tei:div[@type = 'commentary']/
        @subtype"/>
    <xsl:text>)</xsl:text>
  </xsl:if>
  <xsl:text>\end{german}\par\vspace{3mm}</xsl:text>
  \begin{pairs}
    \begin{Leftside}
      \beginnumbering
      \pstart
      \firstlinenum{1}
      \linenumincrement{2}
      \foreignlanguage{greek}{
        <xsl:apply-templates select="child::node()"/>
      }\normalsize
    \pend
    \endnumbering
  \end{Leftside}
  <xsl:apply-templates select="parent::tei:div/
    parent::tei:div/parent::tei:div/tei:div[@type =
    'translation']/tei:p[current()]/@xml:id =
    substring-after(@corresp, '#')"/>
</xsl:template>
```

Das in der vorhergehenden Regel aufgerufene Template für den Text der Übersetzung des Kommentarfragments folgt als nächstes. Es werden die TeX Deklarationen für die rechte Spalte eingefügt.

```
<xsl:template match="tei:p[parent::tei:div[@xml:lang = 'de']]">
  \begin{Rightside}
    \beginnumbering
    \numberlinefalse
    \pstart{
      \foreignlanguage{german}{
```

```

        <xsl:value-of select="text()" />
    }
}
\pend
\endnumbering
\end{Rightside}
\end{pairs}
\Columns
</xsl:template>

```

Die folgende Regel erstellt die Bezeugungsleiste. Dabei taucht das Problem auf, dass es manchmal Zusätze zu den Abkürzungen der Handschriften gibt, z.B.:  $C^{sec}$ . Dieser Fall wird mit `analyze-string` verarbeitet.

```

<xsl:template match="tei:app[@type = 'fragment']">
  \begin{german}
    \textit{<xsl:value-of select="tei:rdg/text()" />}
    <xsl:text> </xsl:text>
    <xsl:for-each select="tokenize(tei:rdg/@wit, ' ')">
      <xsl:analyze-string select="substring-after(., '#')"
        regex="(V1|C|P1|G|O|M|B1|P2|B2|N1|P3|A1|P4|P5|P6|
          P7|P8|L1|Z|V2|V3|V4|V5|A2|A3|N2|L2) (.*)">
        <xsl:matching-substring>
          <xsl:value-of select="regex-group(1)" />
          <xsl:if test="regex-group(2) != ''">
            <xsl:text>\textsuperscript{</xsl:text>
              <xsl:value-of select="regex-group(2)" />
            <xsl:text>}</xsl:text>
          </xsl:if>
        </xsl:matching-substring>
      </xsl:analyze-string>
      <xsl:if test="position() != last()">
        <xsl:text> </xsl:text>
      </xsl:if>
    </xsl:for-each>
  \end{german}\par\vspace{3mm}
</xsl:template>

```

Die nächste Regel erstellt einen Eintrag im kritischen Apparat. Wenn schon ein Eintrag im Apparat vorausgegangen ist, wird ein langer Bindestrich eingefügt.

```

<xsl:template match="tei:app[@type = 'text']" >
  <xsl:if test="(local-name(preceding-sibling::*[1]) = 'app')
    and (preceding-sibling::tei:app/@type = 'text')">
    <xsl:text>--- </xsl:text>
  </xsl:if>
  <xsl:apply-templates select="tei:lem"/>
  <xsl:text>\hspace{-0.75mm}] </xsl:text>
  <xsl:for-each select="tei:rdg">
    <xsl:apply-templates select="tei:foreign | text()" />
    <xsl:text> </xsl:text>
    <xsl:for-each select="tokenize(./@wit, ' ')">
      <xsl:analyze-string select="substring-after(., '#')"

```

```

        regex="(V1|C|P1|G|O|M|B1|P2|B2|N1|P3|A1|P4|P5|
        P6|P7|P8|L1|Z|V2|V3|V4|V5|A2|A3|N2|L2)(.*)">
    <xsl:matching-substring>
        <xsl:value-of select="regex-group(1)"/>
        <xsl:if test="regex-group(2) != ''">
            <xsl:text>\textsuperscript{</xsl:text>
                <xsl:value-of select="regex-group(2)"/>
            <xsl:text>}</xsl:text>
        </xsl:if>
    </xsl:matching-substring>
</xsl:analyze-string>
<xsl:if test="position() != last()">
    <xsl:text> </xsl:text>
</xsl:if>
</xsl:for-each>
</xsl:for-each>
</xsl:template>

```

Es folgen einige Regeln für das Lemma eines kritischen Apparats, für griechischen Text und für Superscript bzw. Italic Text.

```

<xsl:template match="tei:lem">
    <xsl:apply-templates select="child::node()"/>
</xsl:template>

<xsl:template match="tei:foreign[@xml:lang = 'grc']">
    \foreignlanguage{greek}{<xsl:value-of select="./text()"/>}
</xsl:template>

<xsl:template match="tei:hi[@rend = 'sup']">
    <xsl:text>\textsuperscript{</xsl:text>
        <xsl:apply-templates select="child::node()"/>
    <xsl:text>}</xsl:text>
</xsl:template>

<xsl:template match="tei:hi[@rend = 'italic']">
    <xsl:text>\textit{</xsl:text>
        <xsl:apply-templates select="child::node()"/>
    <xsl:text>}</xsl:text>
</xsl:template>

```