

CET User Manual

Version 1.1

Critical Edition Typesetter

A system for typesetting critical editions on PCs

Copyright (C) 1995 Bernt Karasch. All rights reserved. Typeset with CET.

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically those of translation, reprinting, reuse of illustrations, broadcasting, reproduction by photocopying machine or similar means, and storage in data banks.

The use of registered names, trademarks, etc. in the publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

CONTENTS

INTRODUCTION	1
CHAPTER 1 — CET and the CET shell	3
SUMMA.TXT and CET output	8
CHAPTER 2 — Page formatting	10
2.1 Image area	10
2.2 Crop marks	12
2.3 Headline	13
2.4 Hyphenation rules	14
2.5 Line numbering	15
SUMMA.TXT and CET output	18
CHAPTER 3 — Text formatting	20
3.1 Basic formatting commands	20
3.2 Greek text	21
3.3 Typesetting side by side	21
3.4 Typesetting with two columns	22
3.5 Font selection	23
3.6 Margins and indentations	23
3.7 Spaces and page breaks	24
3.8 Including text files	25
3.9 Special commands	25
3.10 Accents and special characters	27
SUMMA.TXT and CET output	28
CHAPTER 4 — Adding fonts	30
CHAPTER 5 — Footnotes, endnotes and filenotes	32
5.1 Footnotes	32
5.2 Endnotes	37
5.3 Filenotes	38
SUMMA.TXT and CET output	40
CHAPTER 6 — References	42
CHAPTER 7 — Indices	44
7.1 Index of words from the main text	44
7.2 Index of words from the variant readings	45
7.3 Index of labels	47
7.4 Index of references	47
CHAPTER 8 — Creating an edition — step by step	48

APPENDIX A — Limitations	49
APPENDIX B — Parameter types	50
APPENDIX C — Solutions to the exercises	54
APPENDIX D — Common problems and solutions	59
APPENDIX E — Error messages	60
APPENDIX F — The authors of the CET components	66
APPENDIX G — Trademarks	68
APPENDIX H — Command names and keywords	69
H.1 Command names	69
H.2 Keywords	71
INDEX	75

INTRODUCTION

This manual describes how to use the *Critical Edition Typesetter* (CET), a system for typesetting critical editions on PCs. By means of exercises you can check your knowledge acquired by reading this manual. After finishing the last exercise you are ready to print your first page of a critical edition typeset with CET.

What CET cannot do for you:

CET is not an automatic manuscript collation program. You have to collate the manuscripts by yourself as before. If you are looking for an automatic manuscript collation program, you can try out *COLLATE*¹.

CET cannot compare different collations.

CET is not a WYSIWYG program (what you see is what you get): The text must be processed by CET in order to see what you get from your input.

What CET can do for you:

CET is a system for typesetting critical editions. CET helps you to create your critical edition by

managing up to nine independent footnote series, which can be printed justified, unjustified, in two or three columns,

computing the line numbers in the margin and the line numbers in front of the lemmata in the critical apparatus,

replacing symbolic references with the correct page and line numbers,

providing the following standard PostScript fonts:

AvantGarde

Bookman

Courier

Helvetica

NewCenturySchlbk

Palatino

Times

ZapfChancery

a Greek font (ΑΒΓ Δαβγδ),

¹ author: Peter Robinson, The Computers and Manuscripts Project, Oxford University Computing Service, 13 Banbury Road, Oxford OX2 6NN, England. Internet: peterr@vax.oxford.ac.uk

supporting any Adobe Type 1 PostScript font,

abbreviating lemmata in the critical apparatus consisting of more than two words (optional),

generating word indices,

generating printouts on almost all printers customary in trade,

and generating a PostScript file of your edition, which can be fed into a PostScript typesetter in the printing office without further intervention by a compositor.

CHAPTER 1 — CET and the CET shell

The development of CET was motivated by PROF. DR. L. HÖDL (Bochum), who was looking for an adequate computer program for typesetting his contribution to the edition of HENRY OF GHENT'S *Summa*. As the word processing programs customary to trade do not support more than one footnote series and do not support the computing of line numbers for the critical apparatus, the only way out was to develop a new program. Fortunately, there already were a lot of powerful programs, which could be integrated into CET.

The kernel of CET is the program \TeX . \TeX was developed by D. E. Knuth (Stanford University) as a program intended primarily for typesetting mathematical texts. \TeX generates a DVI file (DeVice Independent file) from a \TeX input file, which contains the text to be printed along with the formatting commands. This DVI file can be previewed on the screen and can be printed on a broad range of output devices using suitable DVI drivers. The version of \TeX used within CET was ported to the PC architecture by E. Mattes (Universität Stuttgart).

The functionality of \TeX can be improved by so-called macro packages. CET uses the macro packages EDMAC and LaTeX2e. EDMAC was developed by John Lavagnino (Brandeis University) and Dominik Wujastyk (Wellcome Institute for the History of Medicine, London) and adds to \TeX functions for typesetting critical editions. The macro package LaTeX2e provides functions for easy selection of PostScript fonts.

In order to spare you working yourself in with \TeX and in order to provide functionality not available with \TeX alone, CET contains a preprocessor (CEPP.EXE), which converts your input file into a \TeX input file for processing by \TeX . Using the macro packages EDMAC and LaTeX2e \TeX generates a DVI file from the \TeX input file generated by the preprocessor. This DVI file can be previewed on the screen with a program which is part of CET.

As there are no printers, which can print DVI files, we need an additional program: *dvips*, developed by T. Rokicki and D. E. Knuth (Stanford University), converts a DVI file into a PostScript file. This PostScript file can be printed on PostScript output devices (printers and typesetters) without further modification. For printing on printers, which do not understand PostScript, CET uses the PostScript emulator Ghostscript. Ghostscript can preview the PostScript file on the screen but cannot zoom in and zoom out - in contrast to the previewer for DVI files.

The text of the edition must be entered using a word processing program. CET contains the shareware word processing program BOXER, which must be registered if you decide to keep using it after the evaluation period¹. BOXER is one of the few word processing

¹ German source: Nane Jürgensen, Nordergraben 26, 24937 Flensburg, Tel. (0461) 182340, Fax (0461) 182341, CompuServe: 100021,414

programs, which can highlight specific combinations of letters by a different colour (syntax highlighting). CET makes use of this feature by using different colours for text to be printed and for typesetting commands. Instead of BOXER, you can use any word processing program, which can export the text as an ASCII file.

For easy operation all components of CET are combined in the CET menu:

- 1 —> Edit
- 2 —> Typeset
- 3 —> Preview
- 4 —> Print
- 5 —> PostScript-Preview
- 6 —> Settings
- 7 —> Quit

The normal process consists of the cycle Edit - Typeset - Preview. By choosing Print you can print the edition within this cycle at any time. After choosing Settings you can change the layout of your edition. Your edition is saved in a file with the filename extension .TXT. The layout for your edition is saved in a file with the same name, but with the filename extension .CFG. In the following text this file is called *configuration file*.

The following summary explains what happens when you select an item from the CET menu:

- | | |
|---------|---|
| Edit | asks for the name of the text file to be processed and loads it into the word processing program selected during the installation of CET (e. g., BOXER). |
| Typeset | asks for the name of the text file to be processed, converts this text file into two T _E X files (*.TEX and *.STY) using the preprocessor CEPP.EXE and typesets these two T _E X files using the program T _E X and the macro packages EDMAC and LaTeX2e. Should your text file contain an error, CET displays a corresponding error message on the screen using the selected word processing program (see Appendix E). After quitting the word processing program CET loads the faulty text file into the selected word processing program, so that you can correct your mistake. |


```

*.TXT, *.CFG
  |
  | CEPP.EXE
*.TEX, *.STY
  |
  | TEX386.EXE (macro packages: EDMAC 3.16, LaTeX2e)
*.DVI

```

Preview

asks for the name of the file to be previewed and displays the corresponding DVI file, which has been created by the previous call to Typeset. The most important commands within the previewer are:

zoom in (+), zoom out (-), next page (Page ↓), previous page (Page ↑), select a page (p, page number, Return), search for text (s, text, Return), quit (q)

A complete list of commands for the previewer can be found in the file `\EMTEX\DOC\ENGLISH\DVIDRV.DOC` in the section "Keyboard functions for dviscr and dvik".

Print

asks for the name of the file and the numbers of the pages to be printed (examples: empty input prints the whole text, 5 prints page 5 and 5–10 prints the pages 5–10), converts the corresponding DVI file generated by the previous call to Typeset into a PostScript file using the PostScript converter DVIPS32.EXE and copies this PostScript file to the printer (in case of a PostScript output device) or uses the PostScript emulator Ghostscript (GS386.EXE) for printing (in case of an output device, which does not understand PostScript):

```

*.DVI
  |
  | DVIPS32.EXE
*.PS
  |
  | COPY *.PS PRN |
  |               |
  | printout on printers |
  | with             without
  | PostScript

```

PostScript-Preview

asks for the name of the file and the numbers of the pages to be previewed (examples: empty input displays the whole text, 5 displays page 5 and 5–10 displays the pages 5–10), converts the corresponding DVI file generated by the previous call to Typeset into a PostScript file using the PostScript converter DVIPS32.EXE and displays this PostScript file on the screen using the PostScript emulator Ghostscript (GS386.EXE):

```

*.DVI
|
| DVIPS32.EXE
|
*.PS
|
| GS386.EXE
|

```

preview on screen

Settings

asks for the name of the file whose settings are to be changed or are to be displayed and loads the corresponding configuration file (*.CFG) into the selected word processing program (e. g., BOXER).

Quit

quits CET.

Exercise 1:

1. Look at the contents of the file SUMMA.TXT by starting CET, choosing Edit and typing SUMMA (you have to enter the filename only once; afterwards it suffices to accept the default (SUMMA) by pressing Return). Quit the word processing program (BOXER: Alt-X).
2. Typeset the text (Typeset).
3. Preview the text on the screen (Preview and PostScript-Preview).
4. Print the text (Print).
5. Look at the contents of the corresponding configuration file (Settings). Quit the word processing program (BOXER: Alt-X).

Please work through the exercises in the following chapters. After finishing all exercises you can print out your first page of a critical edition typeset with CET. If a chapter contains exercises, you can find a printout of the file SUMMA.TXT and a printout of SUMMA.TXT typeset with CET at the end of the corresponding chapter on two adjacent pages, so that you can check your solutions to the exercises.

This page is blank, so that the two following pages are adjacent.

ARTICULUS XLII
DE DEI PERFECTIIONE

Sequitur de Dei perfectione. Circa quam duo sunt inquirenda: primo, si Deus possit dici perfectus; secundo, si perfectione cuiuslibet creaturae sit perfectus.

QUAESTIO 1

Utrum Deus possit dici perfectus

Circa primum arguitur, quod Deus non possit dici perfectus.

Primo sic. Omne perfectum est factum, quia praepositio illa 'per' in proposito non diminuit, sed potius auget. Deus non est factus, quia per ipsum facta sunt omnia., ut vult Augustinus, sermone Io Super Ioannem. Ergo etc.

Secundo sic. Id, cuius natura stat in eo quod supra se recipit omnes conditiones nobilitatis, maxime imperfectum est, quia de ratione perfecti est quod non recipit additionem in dignitate et nobilitate, et de ratione imperfecti quod recipit additionem, ut iam dicetur. Deus est huiusmodi, quia non est nisi esse purum, ut habitum est supra. Esse autem simpliciter et purum super se recipit omnes determinationes dignitatis et nobilitatis, ut sunt sapientia, bonitas pulchritudo, et cetera huiusmodi. Ergo etc.

Tertio sic. Quae perfecta sunt per se, nihil invenitur extra illa, ut dicit Philosophus, Vo Metaphysicae, cap. De perfecto. Et dicit, Io Caeli et mundi, quod corpus universi ex hoc est perfectum, quia non habet aliud corpus extra, quod tangit ipsum. Extra Deum autem sunt omnes creaturae. Ergo etc.

1

ARTICULUS XLII DE DEI PERFECTIOE

Sequitur de Dei perfectione. Circa quam duo sunt inquirenda: primo, si Deus possit dici perfectus; secundo, si perfectione cuiuslibet creaturae sit perfectus.

QUAESTIO 1

Utrum Deus possit dici perfectus

Circa primum arguitur, quod Deus non possit dici perfectus. Primo sic. Omne perfectum est factum, quia praepositio illa 'per' in proposito non diminuit, sed potius auget. Deus non est factus, quia per ipsum facta sunt omnia., ut vult Augustinus, sermone Io Super Ioannem. Ergo etc. Secundo sic. Id, cuius natura stat in eo quod supra se recipit omnes conditiones nobilitatis, maxime imperfectum est, quia de ratione perfecti est quod non recipit additionem in dignitate et nobilitate, et de ratione imperfecti quod recipit additionem, ut iam dicitur. Deus est huiusmodi, quia non est nisi esse purum, ut habitum est supra. Esse autem simpliciter et purum super se recipit omnes determinationes dignitatis et nobilitatis, ut sunt sapientia, bonitas pulchritudo, et cetera huiusmodi. Ergo etc. Tertio sic. Quae perfecta sunt per se, nihil invenitur extra illa, ut dicit Philosophus, Vo Metaphysicae, cap. De perfecto. Et dicit, Io Caeli et mundi, quod corpus universi ex hoc est perfectum, quia non habet aliud corpus extra, quod tangit ipsum. Extra Deum autem sunt omnes creaturae. Ergo etc.

CHAPTER 2 — Page formatting

This chapter explains how you can change the page layout of your edition. Typical settings you may want to change are: the width and the height of the text, the default font, the space between words and the appearance of the headline.

To change a setting in the configuration file proceed as follows:

Call the CET menu item Settings. CET loads the configuration file into the selected word processing program. Search for the keyword corresponding to the setting you are going to change and change the settings at the right side of the keyword according to your needs. Quit the word processing program (BOXER: Alt-X, W) and call the CET menu items Typeset and Preview in order to see the effect of your changes to the configuration file.

NOTE: If a configuration file does not exist, CET creates a new configuration file with default values during the first call to Typeset.

NOTE: Throughout the manual the default value is listed along with the parameter type within brackets after the keyword:

keyword (default value, *parameter type*)

The parameter types are listed in Appendix B.

Example: changing the page width to 12 cm

Call Settings. If an empty file is displayed, a configuration file does not exist. In this case quit the word processing program (BOXER: Alt-X) and call Typeset. Call Settings a second time. Search the configuration file for the keyword `HORIZONTAL_SIZE`. Change the default value (11.0 cm) to 12.0 cm. Quit the word processing program (BOXER: Alt-X, W) and call Typeset and Preview in order to see the effect of your changes to the configuration file.

1. Image area

`HORIZONTAL_SIZE` (11.0 cm, *value*) defines the width of the text without the line numbers in the margin. `VERTICAL_SIZE` (17.0 cm, *value*) defines the height of the text without the headline.

`HORIZONTAL_OFFSET` (-0.3 cm, *value*) and `VERTICAL_OFFSET` (1.5 cm, *value*) define the horizontal and vertical shift of the printout on the paper respectively. Negative values cause a shift to the left and to the top of the paper respectively, whereas positive values cause a shift to the right and to the bottom of the paper respectively. These settings may be used for centering the printout on the paper.

HORIZONTAL_TOLERANCE (0.1 pt, *value*) defines to which extent the width of a line may go beyond the width defined with HORIZONTAL_SIZE. If and only if MARK_BAD_LINES (N *Y, *yes/no*) is active (N *Y), lines wider than HORIZONTAL_SIZE + HORIZONTAL_TOLERANCE are marked with a black rectangle in the right margin. Appendix D contains information how to avoid these black rectangles.

GLOBAL_FONT (Times m m n 12.0 pt 12.0 pt, *font*) defines the font used unless CET encounters a new font definition in the text (<[...]>, see 3.5 Font selection).

LATIN_FONT (Times m m n 12.0 pt 12.0 pt, *font*) defines the font used if you use <LTB>...<LTE> to switch to Latin text from within non-Latin text (Greek, Hebrew, Arabic; see 3.1 Basic formatting commands).

PARAGRAPH_INDENTATION (1.0 cm, *value*) defines the paragraph indentation (the indentation of the first line of a paragraph).

If FRENCH_SPACING (N *Y, *yes/no*) is active (N *Y), the space at the end of a sentence is as wide as the space between words. If FRENCH_SPACING is inactive (*N Y), the space at the end of a sentence is a bit wider than the space between words.

The space between letters spaced out with <SPB>...<SPE> (see 3.1 Basic formatting commands) can be defined with SPACE_OUT_DISTANCE (0.2 em, *value*).

WORD_GLUE (0.3 em plus 0.2 em minus 0.1 em, *space*) defines the normal, minimal and maximal space between words within the main text (i. e., all words outside the critical apparatus). The default values define the normal space between words as 0.3 em, the minimal space as 0.3 em - 0.1 em = 0.2 em and the maximal space as 0.3 em + 0.2 em = 0.5 em. APPARATUS_WORD_GLUE (0.3 em plus 0.2 em minus 0.1 em, *space*) defines the space between words within the critical apparatus and FOOTNOTE_PARAGRAPH_GLUE (1.0 em plus 0.4 em minus 0.4 em, *space*) defines the space between justified lemmata in the critical apparatus (see 5.1 Footnotes).

If the text of a lemma appears more than once within a specific line, CET adds a running exponent to the corresponding lemmata so that the reader can attach the lemmata in the critical apparatus to the right places in the text. WORD_DISTANCE (20, *value*) defines the number of words, which CET looks to right and to the left of each lemma in order to check whether the lemma is unique in its line. Therefore, the number defined with WORD_DISTANCE must be greater than the maximal number of words within a printed line. The automatic addition of exponents can be switched off with WORD_DISTANCE 0.

Example:

1 C A C

1 C ²] variant reading

The variant reading "variant reading" belongs to the second C in line 1. In this case CET automatically generates the exponent 2 in the critical apparatus, if the number defined with `WORD_DISTANCE` is at least 2. If the number is less than 2 the exponent is not printed. The greater the number defined with `WORD_DISTANCE`, the longer it takes to process your text. Thus it is recommended to suppress the automatic generation of exponents during the revisions of the text with `WORD_DISTANCE 0` and to activate the generation of exponents only if you plan to print out your current revision.

Exercise 2:

Apply the following settings to the text (read the information about Font and Space in appendix B):

Image area:	130 mm x 190 mm
default font:	Times 12 point, leading 14 point
paragraph indentation:	5 mm
distance between words:	normal 0.4 em, minimal 0.1 em, maximal 0.6 em

Typeset and Print the text with the new settings. Adjust `HORIZONTAL_OFFSET` and `VERTICAL_OFFSET`, so that the printout is centered on the paper.

2. Crop marks

Crop marks help the printing office with cutting the printed sheets into pages. Ask your printing office whether crop marks are required or not. If your printing office asks for crop marks, it can tell you the required values, which are explained in the following text.

`CROP_MARK_WIDTH` (0.4 pt, *value*) defines the thickness of the lines of the crop marks. The printing of crop marks can be switched off with `CROP_MARK_WIDTH 0` pt.

`CROP_MARK_GAP` (5.0 pt, *value*) defines the space between the vertical and the horizontal line of a crop mark.

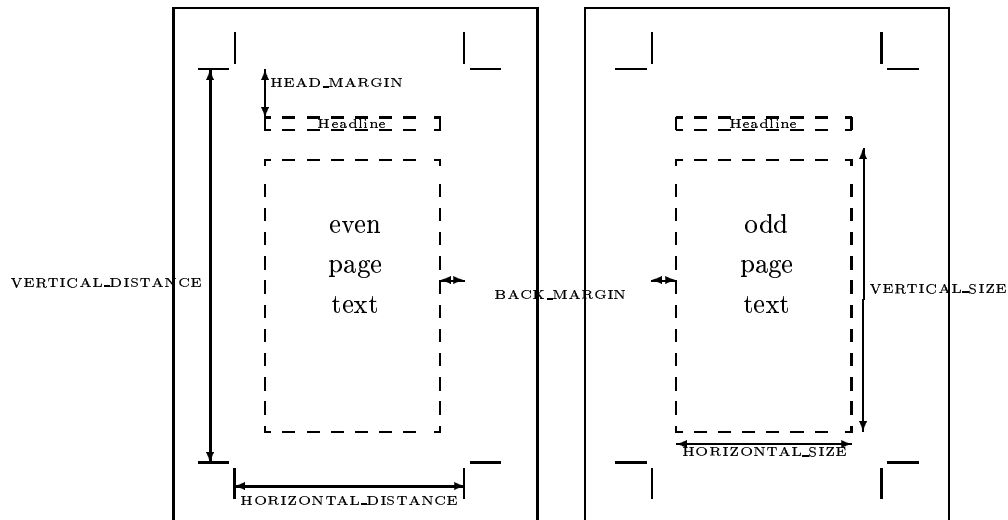


Figure 1: crop marks

The meaning of the following settings can be gathered from Figure 1:

CROP_MARK_HORIZONTAL_DISTANCE	(16.1 cm, <i>value</i>)
CROP_MARK_VERTICAL_DISTANCE	(23.4 cm, <i>value</i>)
CROP_MARK_HEAD_MARGIN	(1.9 cm, <i>value</i>)
CROP_MARK_BACK_MARGIN	(2.5 cm, <i>value</i>)

Exercise 3:

Adjust the crop marks to the following values:

thickness:	0.5 point
space:	4.0 point
horizontal distance:	16 cm
vertical distance:	25 cm
head margin:	2 cm
back margin:	2 cm

3. Headline

The headline contains the page number and the running header.

HEADLINE_VERTICAL_DISTANCE (0.4 cm, *value*) defines the vertical distance between the headline and the text.

The position of the running header within the headline can be defined with HEADLINE_POSITION (inner *center outer, *selection*), the font for the running header can be defined with HEADLINE_FONT (Times m m n 10.0 pt 10.0 pt, *font*).

Defaults for the running headers on left (even) and right (odd) pages can be defined with HEADLINE_TEXT_LEFT (, *text*) and HEADLINE_TEXT_RIGHT (, *text*) respectively.

The running headers can be changed from within the text. <HLB>*new running header*<HLE> causes CET to put "*new running header*" in the headline of all following left (even) pages. The running header of all following right (odd) pages can be changed from within the text with <HRB>...<HRE>.

HEADLINE_TEXTL_BEGIN	(<HLB>, <i>command name</i>)
HEADLINE_TEXTL_END	(<HLE>, <i>command name</i>)
HEADLINE_TEXTR_BEGIN	(<HRB>, <i>command name</i>)
HEADLINE_TEXTR_END	(<HRE>, <i>command name</i>)

The font for the page numbers can be defined with PAGE_NUMBER_FONT (Times m m n 10.0 pt 10.0 pt, *font*). The page numbering can be changed from within the text: <PNB>12<PNE> causes CET to set the current page number to 12, so that the current page is printed with page number 12, the following one with page number 13 and so on.

PAGE_NUMBER_BEGIN	(<PNB>, <i>command name</i>)
PAGE_NUMBER_END	(<PNE>, <i>command name</i>)

Exercise 4:

Adjust the configuration file to the following settings:

distance headline <—> text:	0.75 cm
position of running header:	inner
font for the running header:	Times 10 point, leading 12 point
running header on left pages:	ARTICULUS XLII
running header on right pages:	QUAESTIO 1
page number:	18
font for the page number:	Times 12 point, leading 14 point

4. Hyphenation rules

CET contains hyphenation rules for German, French, Latin and English. HYPHENATION_RULES (German French *Latin English None Spare1 Spare2, *selection*) activates hyphenation rules for the corresponding language. If you select None, CET does not hyphenate at all. Spare1 und Spare2 are reserved and are equivalent to None.

You can switch between the different languages from within the text with <GR>, <FR>, <LAT>, <UK>, <NON>, <SP1>, <SP2>.

Example:

```
<GR>

paragraph 1

<UK>

paragraph 2
```

Words in paragraph 1 are hyphenated according to German hyphenation rules, whereas words in paragraph 2 and in all following paragraphs are hyphenated according to English hyphenation rules.

HYPHENATION_GERMAN	(<GR>, <i>command name</i>)
HYPHENATION_FRENCH	(<FR>, <i>command name</i>)
HYPHENATION_LATIN	(<LAT>, <i>command name</i>)
HYPHENATION_ENGLISH	(<UK>, <i>command name</i>)
HYPHENATION_NONE	(<NON>, <i>command name</i>)
HYPHENATION_SPARE1	(<SP1>, <i>command name</i>)
HYPHENATION_SPARE2	(<SP2>, <i>command name</i>)

Exercise 5:

Activate Latin hyphenation rules at the beginning of the text.

5. Line numbering

This section explains how you can modify the appearance of the line numbers in the margin.

LINEATION_BY (page *section, *selection*) defines the lineation mode: lineation by page (the first line of each page has the line number 1) or lineation by section (lines within #N+ ... #N- and simultaneously within #L+ ... #L- are numbered sequentially regardless of page breaks).

The position of the line numbers in the margin can be defined with LINEATION_MARGIN (left right *inner outer, *selection*), the font for the line numbers in the margin can be defined with LINE_NUMBER_FONT (Times m m n 10.0 pt 10.0 pt, *font*) and the distance between the line numbers in the margin and the text can be defined with LINE_NUMBER_DISTANCE (1.0 pc, *value*).

The first line number to be printed can be defined with LINE_NUMBER_FIRST (5, *number*), the distance between line numbers to be printed can be defined with LINE_NUMBER_INCREMENT (5, *number*). The corresponding keywords for subline numbers are SUB_LINE_NUMBER_FIRST (5, *number*) and SUB_LINE_NUMBER_INCREMENT (5, *number*).

The line numbering must be started with `#N+`. This command resets the line number counter to 0. Now the printing of line numbers can be started with `#L+` and can be stopped with `#L-`. The corresponding commands for subline numbers are `#S+` and `#S-`. The end of a numbered section must be marked with `#N-`. The commands `#N+`, `#L+`, `#L-` and `#N-` relate to the paragraph following the respective command.

<code>NUMBERING_BEGIN</code>	<code>(#N+, command name)</code>
<code>NUMBERING_END</code>	<code>(#N-, command name)</code>
<code>LINEATION_BEGIN</code>	<code>(#L+, command name)</code>
<code>LINEATION_END</code>	<code>(#L-, command name)</code>
<code>SUB_LINEATION_BEGIN</code>	<code>(#S+, command name)</code>
<code>SUB_LINEATION_END</code>	<code>(#S-, command name)</code>

Example:

```
#N+ #L+
paragraph 1
#L-
paragraph 2
#L+
paragraph 3
#L- #N-
#N+ #L+
paragraph 4
#L- #N-
```

in combination with the settings

```
LINE_NUMBER_FIRST 1
LINE_NUMBER_INCREMENT 1
```

results in:

```
1 paragraph 1
  paragraph 2
2 paragraph 3
1 paragraph 4
```

You can freeze the current line number with `#K+` and you can switch off the freezing of the line number with `#K-`. `DISPLAY_LOCKED_LINE_NUMBER` (`*first last all, selection`) defines whether a frozen line number is printed in the first, last or in all lines.

<code>LINEATION_LOCK</code>	<code>(#K+, command name)</code>
<code>LINEATION_UNLOCK</code>	<code>(#K-, command name)</code>

If CUT_LINE_NUMBER (*N Y, *yes/no*) is active (N *Y), only the last two digits of the line numbers in the margin are printed. The corresponding keyword for subline numbers is CUT_SUBLINE_NUMBER (*N Y, *yes/no*).

Exercise 6:

Modify the configuration file and the text, so that all lines get sequential line numbers. The line numbering is to restart with 1 in the line containing QUAESTIO 1.

<PNB>18<PNE>

<LAT>

#N+ #L+

ARTICULUS XLII
DE DEI PERFECTIONE

Sequitur de Dei perfectione. Circa quam duo sunt inquirenda: primo, si Deus possit dici perfectus; secundo, si perfectione cuiuslibet creaturae sit perfectus.

#L- #N-

#N+ #L+

QUAESTIO 1

Utrum Deus possit dici perfectus

Circa primum arguitur, quod Deus non possit dici perfectus.

Primo sic. Omne perfectum est factum, quia praepositio illa 'per' in proposito non diminuit, sed potius auget. Deus non est factus, quia per ipsum facta sunt omnia., ut vult Augustinus, sermone Io Super Ioannem. Ergo etc.

Secundo sic. Id, cuius natura stat in eo quod supra se recipit omnes condiciones nobilitatis, maxime imperfectum est, quia de ratione perfecti est quod non recipit additionem in dignitate et nobilitate, et de ratione imperfecti quod recipit additionem, ut iam dicetur. Deus est huiusmodi, quia non est nisi esse purum, ut habitum est supra. Esse autem simpliciter et purum super se recipit omnes determinationes dignitatis et nobilitatis, ut sunt sapientia, bonitas pulchritudo, et cetera huiusmodi. Ergo etc.

Tertio sic. Quae perfecta sunt per se, nihil invenitur extra illa, ut dicit Philosophus, Vo Metaphysicae, cap. De perfecto. Et dicit, Io Caeli et mundi, quod corpus universi ex hoc est perfectum, quia non habet aliud corpus extra, quod tangit ipsum. Extra Deum autem sunt omnes creaturae. Ergo etc.

#L- #N-

ARTICULUS XLII DE DEI PERFECTIOE

Sequitur de Dei perfectione. Circa quam duo sunt inquirenda: primo, si
Deus possit dici perfectus; secundo, si perfectione cuiuslibet creaturae sit
perfectus.

QUAESTIO 1

Utrum Deus possit dici perfectus

Circa primum arguitur, quod Deus non possit dici perfectus. Primo sic.
Omne perfectum est factum, quia praepositio illa 'per' in proposito non
diminuit, sed potius auget. Deus non est factus, quia per ipsum facta sunt
omnia., ut vult Augustinus, sermone Io Super Ioannem. Ergo etc. Secundo
sic. Id, cuius natura stat in eo quod supra se recipit omnes condiciones
nobilitatis, maxime imperfectum est, quia de ratione perfecti est quod non
recipit additionem in dignitate et nobilitate, et de ratione imperfecti quod
recipit additionem, ut iam dicitur. Deus est huiusmodi, quia non est nisi
esse purum, ut habitum est supra. Esse autem simpliciter et purum super
se recipit omnes determinationes dignitatis et nobilitatis, ut sunt sapientia,
bonitas pulchritudo, et cetera huiusmodi. Ergo etc. Tertio sic. Quae perfecta
sunt per se, nihil invenitur extra illa, ut dicit Philosophus, Vo Metaphysicae,
cap. De perfecto. Et dicit, Io Caeli et mundi, quod corpus universi ex hoc
est perfectum, quia non habet aliud corpus extra, quod tangit ipsum. Extra
Deum autem sunt omnes creaturae. Ergo etc.

CHAPTER 3 — Text formatting

This chapter explains the basic formatting commands. The most important formatting commands are: font selection, font mark-up (normal, italic, bold, slanted, small capitals), alignment (left-aligned, right-aligned, centered), margins, horizontal and vertical spaces, two-column printing and printing side by side.

1. Basic formatting commands

The end of a paragraph is marked by one or more empty lines.

Several spaces, which are connected but are not spaced out, are interpreted as one space (i. e., a normal space between words).

	input	printout	keyword	default
hyphen	-	-		
hyphen (from-to)	--	—		
dash	---	---		
quotation marks	< > << >> ,, “	< > « » „ “		
italic	in <IB>eo<IE> quod	in <i>eo</i> quod	ITALIC_BEGIN/END	<IB>/<IE>
normal	<IB>in <NB>eo<NE> quod<IE>	in <i>eo</i> quod	NORMAL_BEGIN/END	<NB>/<NE>
upright italic ¹	in <UIB>eo<UIE> quod	in <i>eo</i> quod	UPRIGHT_ITALIC_BEGIN/END	<UIB>/<UIE>
slanted	in <SB>eo<SE> quod	in <i>eo</i> quod	SLANTED_BEGIN/END	<SB>/<SE>
small capitals	in <SCB>eo<SCE> quod	in <i>EO</i> quod	SMALL_CAPS_BEGIN/END	<SCB>/<SCE>
spaced out	in <SPB>eo<SPE> quod	in <i>e o</i> quod	SPACED_BEGIN/END	<SPB>/<SPE>
underlined	in <UB>eo<UE> quod	in <u><i>eo</i></u> quod	UNDERLINED_BEGIN/END	<UB>/<UE>
exponents	in <RB>eo<RE> quod	in ^{eo} quod	RAISED_BEGIN/END	<RB>/<RE>
indices	in <LB>eo<LE> quod	in _{eo} quod	LOWERED_BEGIN/END	<LB>/<LE>
bold	in <BDB>eo<BDE> quod	in eo quod	BOLD_BEGIN/END	<BDB>/<BDE>
centered	<CB>in <i>eo</i> quod<CE>	in <i>eo</i> quod	CENTERED_BEGIN/END	<CB>/<CE>
left-aligned	<LAB>in <i>eo</i> quod<LAE>	in <i>eo</i> quod	LEFT_BEGIN/END	<LAB>/<LAE>
right-aligned	<RAB>in <i>eo</i> quod<RAE>	in <i>eo</i> quod	RIGHT_BEGIN/END	<RAB>/<RAE>
Greek	in <GB>eo<GE> quod	in <i>eo</i> quod	GREEK_BEGIN/END	<GB>/<GE>
Latin	<GB>in <LTB>eo<LTE> quod<GE>	in <i>eo</i> quod	LATIN_BEGIN/END	<LTB>/<LTE>

Exercise 7:

Format the text according to the following items:

end of paragraph:	ARTICULUS XLII; DE DEI PERFECTIOE; sit perfectus.; QUAESTIO 1; UTRUM DEUS POSSIT DICI PERFECTUS; dici perfectus.; I o a n n e m . Ergo etc.; huiusmodi. Ergo etc.; creaturae. Ergo etc.
centered:	ARTICULUS XLII; DE DEI PERFECTIOE; QUAESTIO 1; UTRUM DEUS POSSIT DICI PERFECTUS
small capitals:	UTRUM DEUS POSSIT DICI PERFECTUS; AUGUSTINUS; PHILOSOPHUS
spaced out:	Super Ioannem; Metaphysicae; Caeli et mundi
italic:	<i>non est factus, quia per ipsum facta sunt omnia.</i>
quotation marks:	« <i>non est factus, quia per ipsum facta sunt omnia.</i> »
exponents:	I ^o Super Ioannem; V ^o Metaphysicae; I ^o Caeli et mundi

¹ not implemented

2. Greek text

The following conversion tables must be used for Greek text:

alphabet:

```
input   a b g d e z h j i k l m n x o p r s t u f q y w
printout α β γ δ ε ζ η θ ι κ λ μ ν ξ ο π ρ σ τ υ φ χ ψ ω
```

punctuation marks:

```
input   . , ; : ! ? " ' ( )
printout . , ; : ! ; ' « »
```

CET puts a terminal sigma (ς) at the end of words ending with sigma. A normal sigma (σ) is available by entering c instead of s.

accents (must be placed in front of the corresponding letter):

```
acute accent  ´
grave accent  `
circumflex   ~
```

spiritus (must be placed in front of the corresponding letter):

```
asper <
lenis >
```

iota subscriptum (must be placed after the corresponding letter):

```
|
```

diaeresis (must be placed in front of the corresponding letter):

```
"
```

Example:

```
<GB> >en arq~h| >~hn <o l'ogós. <GE>
```

ἐν ἀρχῇ ἦν ὁ λόγος.

The Greek font can be selected with <GB>...<GE>. In addition, the Greek font can be selected under the name LevyGreek just like the standard PostScript fonts (see 3.5 Font selection).

3. Typesetting side by side

Using the commands for printing text side by side you can put two to nine columns side by side. You have to tell CET the width of each column by specifying the percentage of the page width the columns cover within <SWB>...<SWE>. Then you can enter the paragraphs to be put side by side:

```
<SYB>text column 1<SYS>...<SYS>text column n<SYE>
```


5. Font selection

You select a specific font by typing its font specification (see Appendix B, Font) within `<[...]>`. If the font selection command is in a paragraph of its own (i. e., the font selection command is surrounded by spaces and is the only text within the paragraph), the new font is active up to the end of the text or till CET encounters another font selection command. A font selection command within a paragraph containing further text is active from the place of the font selection command up to the end of the paragraph.

```
FONT_BEGIN           (<[, command name)
FONT_END             (]), command name)
```

Example:

```
<[Times m m i t 10 pt 12 pt]>
paragraph 1
paragraph 2
<[- - - sc 10 pt 12 pt]> paragraph 3
paragraph 4
```

results in:

```
paragraph 1
paragraph 2
PARAGRAPH 3
paragraph 4
```

Exercise 8:

Select a font size of 16 point with a leading of 18 point for the headline (ARTICULUS XLII; DE DEI PERFECTIONE) and a font size of 12 point with a leading of 14 point for the remaining text.

6. Margins and indentations

The left margin of the text can be specified with `<LMB>value<LME>`. This paragraph has been formatted with `<LMB>2 cm<LME>`.

```
LEFT_MARGIN_BEGIN   (<LMB>, command name)
LEFT_MARGIN_END     (<LME>, command name)
```

The right margin of the text can be specified with `<RMB>value<RME>`. This paragraph has been formatted with `<RMB>2 cm<RME>`.

```
RIGHT_MARGIN_BEGIN  (<RMB>, command name)
RIGHT_MARGIN_END    (<RME>, command name)
```

The paragraph indentation (the indentation of the first line of a paragraph) can be specified with `<PIB>value<PIE>`. This paragraph has been formatted with `<PIB>2cm<PIE>`.

```
INDENT_BEGIN           (<PIB>, command name)
INDENT_END             (<PIE>, command name)
```

The indentation of the first n or the last lines except the first n lines of a paragraph can be specified with `<HIB>value<HIE>`. n must be specified with `<HAB>number<HAE>`. A positive number causes an indentation of the last lines except the first n lines of the paragraph, whereas a negative number causes an indentation of the first $-n$ lines of the paragraph. This paragraph has been formatted with `<HIB>2cm<HIE><HAB>-2<HAE>`.

```
HANGINDENT_BEGIN      (<HIB>, command name)
HANGINDENT_END        (<HIE>, command name)
HANGAFTER_BEGIN       (<HAB>, command name)
HANGAFTER_END         (<HAE>, command name)
```

7. Spaces and page breaks

The vertical distance between paragraphs can be specified with `<PSB>value<PSE>`.

A horizontal space can be generated with `<HSB>value<HSE>`, a vertical space can be generated with `<VSB>value<VSE>`. \TeX may change (or even remove) these spaces in order to improve the line and page breaks. You can specify a vertical space which \TeX does not change with `<SVSB>value<SVSE>`.

A new page can be started with `<NP>`.

```
PARAGRAPH_SKIP_BEGIN  (<PSB>, command name)
PARAGRAPH_SKIP_END    (<PSE>, command name)
HSKIP_BEGIN           (<HSB>, command name)
HSKIP_END             (<HSE>, command name)
VSKIP_BEGIN           (<VSB>, command name)
VSKIP_END             (<VSE>, command name)
STATIC_VSKIP_BEGIN    (<SVSB>, command name)
STATIC_VSKIP_END      (<SVSE>, command name)
NEW_PAGE              (<NP>, command name)
```

Exercise 9:

Add a vertical space of 0.7 cm after the headline (ARTICULUS XLII; DE DEI PERFECTIONE); add a vertical space of 0.4 cm in front of and after QUAESTIO 1 and after UTRUM DEUS POSSIT DICI PERFECTUS.

8. Including text files

More complex editions consist of several sections (chapters, quaestiones, ...). It is recommended to put each section into its own text file. CET supports this approach by providing a command for including a text file at the current position. The main text file of your edition (e. g., BOOK.TXT) contains the commands for including the several chapters of your edition (e. g., chapter1.TXT, chapter2.TXT, ...). The filenames of the files to be included must be specified within `<IFB>...<IFE>`. The filename may contain a path and must include the filename extension. The `<IFB>...<IFE>` command must be put into a paragraph of its own (i. e., must be surrounded by empty lines). A file being processed due to a `<IFB>...<IFE>` command must not contain any further `<IFB>...<IFE>` commands.

```
INCLUDE_FILE_BEGIN      (<IFB>, command name)
INCLUDE_FILE_END        (<IFE>, command name)
```

Example:

```
----- BOOK.TXT -----
<C><IFB>chapter1.txt<IFE>

<C><IFB>chapter2.txt<IFE>

<IFB>chapter3.txt<IFE>
-----
```

Using `<C>` (see 3.9 Special commands) you can comment out chapters already finished. This saves computing time. Do not forget to remove all `<C>` before typesetting the final version of your edition.

9. Special commands

CET does not interpret formatting commands within `<TB>...<TE>`, but prints them as text. Using this feature you can print CET command names.

```
TRANSPARENT_BEGIN      (<TB>, command name)
TRANSPARENT_END        (<TE>, command name)
```

Only text within `<TMB>...<TME>` may contain \TeX commands. This makes sure that you do not enter the character `\` by mistake. This character would cause an error message during the processing of the text by \TeX .

```
TEX_MODE_BEGIN         (<TMB>, command name)
TEX_MODE_END           (<TME>, command name)
```

The hyphenation algorithm of \TeX is quite good, but not perfect. Should CET hyphenate a word at a wrong place or not at all, you can mark the possible places for a hyphen within the word with `|-`. CET hyphenates words containing `|-` at these places only. Therefore, you should specify all possible places for a hyphen within a word.

HYPHENATION (|-, *command name*)

Comments are texts, which are ignored by CET. CET interprets text after `<C>` as a comment and ignores the following text up to the end of the current paragraph.

COMMENT (<C>, *command name*)

The separator between lemma and variant reading in the critical apparatus (`]`) can be suppressed by putting `<!>` into the variant reading. By putting `<OM>` into the variant reading you can instruct CET to omit the separator and to put an italic *om.* into the printout at the position of the command `<OM>`.

OMIT_SEPARATOR (<!>, *command name*)
 OMIT (<OM>, *command name*)

Example:

```
(*lemma 1*) {variant reading 1}
(*lemma 2*) {<!>variant reading 2}
(*lemma 3*) {<OM> variant reading 3}
```

10. Accents and special characters

name	input	printout	keyword	default
grave accent	<'>a	à	GRAVE	<'>
acute accent	<'>a	á	ACUTE	<'>
circumflex	<^>a	â	CIRCUMFLEX	<^>
diaeresis	<">a	ä	DIERESIS	<">
tilde	<~>a	ã	TILDE	<~>
macron	<=>a	ā	MACRON	<=>
point accent	<.>a	â	DOT	<.>
semicircle	<u>a	ă	BREVE	<u>
háček	<v>a	ǎ	CHECK	<v>
double acute accent	<H>a	ű	HUNGARUMLAUT	<H>
cedilla	<c>a	ç	CEDILLA	<c>
point accent below	<d>a	ạ	DOT_UNDER	<d>
macron below	a	ạ	BAR_UNDER	
ligature OE	<OE>	Œ	CAPITAL_OE	<OE>
ligature oe	<oe>	œ	SMALL_OE	<oe>
ligature AE	<AE>	Æ	CAPITAL_AE	<AE>
ligature ae	<ae>	æ	SMALL_AE	<ae>
A with circle	<Ao>	Å	CAPITAL_A_CIRCLE	<Ao>
a with circle	<ao>	å	SMALL_A_CIRCLE	<ao>
O with slash	<O/>	Ø	CAPITAL_O_SLASH	<O/>
o with slash	<o/>	ø	SMALL_O_SLASH	<o/>
L with slash	<L/>	Ł	CAPITAL_L_SLASH	<L/>
l with slash	<l/>	ł	SMALL_L_SLASH	<l/>
sz	<sz>	ß	SZ	<sz>
section mark	<S/>	§	SECTION	<S/>
i without point	<i>	ı	DOTLESS_I	<i>

<PNB>18<PNE>

<LAT>

#N+ #L+

<[- - - 16 pt 18 pt]>

<CB>ARTICULUS XLII<CE>

<CB>DE DEI PERFECTIONE<CE>

<VSB>0.7 cm<VSE>

<[- - - 12 pt 14 pt]>

Sequitur de Dei perfectione. Circa quam duo sunt inquirenda: primo, si Deus possit dici perfectus; secundo, si perfectione cuiuslibet creaturae sit perfectus.

#L- #N-

#N+ #L+

<VSB>0.4 cm<VSE>

<CB>QUAESTIO 1<CE>

<VSB>0.4 cm<VSE>

<CB><SCB>Utrum Deus possit dici perfectus<SCE><CE>

<VSB>0.4 cm<VSE>

Circa primum arguitur, quod Deus non possit dici perfectus.

Primo sic. Omne perfectum est factum, quia praepositio illa 'per' in proposito non diminuit, sed potius auget. Deus <<<IB>non est factus, quia per ipsum facta sunt omnia.<IE>>>, ut vult <SCB>Augustinus<SCE>, sermone I<RB>o<RE> <SPB>Super Ioannem.<SPE> Ergo etc.

Secundo sic. Id, cuius natura stat in eo quod supra se recipit omnes conditiones nobilitatis, maxime imperfectum est, quia de ratione perfecti est quod non recipit additionem in dignitate et nobilitate, et de ratione imperfecti quod recipit additionem, ut iam dicetur. Deus est huiusmodi, quia non est nisi esse purum, ut habitum est supra. Esse autem simpliciter et purum super se recipit omnes determinationes dignitatis et nobilitatis, ut sunt sapientia, bonitas pulchritudo, et cetera huiusmodi. Ergo etc.

Tertio sic. Quae perfecta sunt per se, nihil invenitur extra illa, ut dicit <SCB>Philosophus<SCE>, V<RB>o<RE> <SPB>Metaphysicae,<SPE> cap. De perfecto. Et dicit, I<RB>o<RE> <SCB>Caeli et mundi,<SCE> quod corpus universi ex hoc est perfectum, quia non habet aliud corpus extra, quod tangit ipsum. Extra Deum autem sunt omnes creaturae. Ergo etc.

#L- #N-

ARTICULUS XLII
DE DEI PERFECTIOE

Sequitur de Dei perfectione. Circa quam duo sunt inquirenda: primo, si Deus possit dici perfectus; secundo, si perfectione cuiuslibet creaturae sit perfectus.

QUAESTIO 1

UTRUM DEUS POSSIT DICI PERFECTUS

Circa primum arguitur, quod Deus non possit dici perfectus.

Primo sic. Omne perfectum est factum, quia praepositio illa 'per' in proposito non diminuit, sed potius auget. Deus *«non est factus, quia per ipsum facta sunt omnia.»*, ut vult AUGUSTINUS, sermone I° Super Ioannem. Ergo etc.

Secundo sic. Id, cuius natura stat in eo quod supra se recipit omnes conditiones nobilitatis, maxime imperfectum est, quia de ratione perfecti est quod non recipit additionem in dignitate et nobilitate, et de ratione imperfecti quod recipit additionem, ut iam dicitur. Deus est huiusmodi, quia non est nisi esse purum, ut habitum est supra. Esse autem simpliciter et purum super se recipit omnes determinationes dignitatis et nobilitatis, ut sunt sapientia, bonitas pulchritudo, et cetera huiusmodi. Ergo etc.

Tertio sic. Quae perfecta sunt per se, nihil invenitur extra illa, ut dicit PHILOSOPHUS, V° Metaphysicae, cap. De perfecto. Et dicit, I° Caeli et mundi, quod corpus universi ex hoc est perfectum, quia non habet aliud corpus extra, quod tangit ipsum. Extra Deum autem sunt omnes creaturae. Ergo etc.

CHAPTER 4 — Adding fonts

CET supports the standard PostScript fonts

AvantGarde

Bookman

Courier

Helvetica

NewCenturySchlbk

Palatino

Times

ZapfChancery

and the Greek font LevyGreek (ΑΒΓΔαβγδ).

Any Adobe Type 1 PostScript font with AdobeStandardEncoding can be installed with the program addpsfnt. The program addpsfnt needs the AFM file (Adobe Font Metric) and the PFB file (PostScript Font Binary) of the fonts to be installed.

You usually do not install a single font but a font family. A font family consists of four fonts: normal, *italic*, **bold** and ***bold-italic***

Installation of a PostScript family:

1. Set your working directory to the directory containing the AFM and the PFB files of the fonts to be installed.
2. Make sure that the AFM files and the PFB files are not read-only (ATTRIB -R *.AFM and ATTRIB -R *.PFB remove the read-only attributes). Note that addpsfnt converts the AFM files to the MS-DOS format: LF, CR and LFCR are converted into CRLF. The PFB files are not changed by addpsfnt.
3. Call addpsfnt with the filenames of the fonts to be installed:

```
addpsfnt <name normal> <name italic> <name bold> <name bold-italic>
```

If there is no italic, bold or bold-italic font you have to specify - for the missing font.

Examples:

```
addpsfnt bm bmi bmb bmbi
```

```
addpsfnt parkave - - -
```

After the installation of the new font family addpsfnt displays the name of the new font family on the screen. Under this name you can use the new font family as if you were using one of the standard PostScript fonts (see 3.5 Font selection). The new font family is available in the following mark-ups: normal, italic, slanted and small capitals

Please use professional PostScript fonts only. This ensures that all characters used by CET are defined in the corresponding PFB files.

Error messages:

message: ; expected: ! syntax error

C 175 ; WX 564 ; N ; B 158 506 158 506 ;

ERROR: *.AFM is bad.

remedy: The AFM file does not contain a name for character 175. Remove the offending line from the AFM file and call addpsfnt again.

message: ERROR: Reading from *.AFM (try again after ATTRIB -R *.AFM)

ERROR: Reading from *.PFB (try again after ATTRIB -R *.PFB)

remedy: 1. The AFM files and PFB files must not be write-protected. Remove the read-only attribute by following the instructions given in the error message and call addpsfnt again.

or

2. The file does not exist in the current working directory.

message: B expected: ! syntax error

C 32 ; WX 425 ; N space ;

ERROR: *.AFM is bad.

remedy: Not available. The AFM file contains an error and cannot be processed by addpsfnt.

message: ; expected: ! syntax error

C 32; WX 0; N space; B 0 0 0 0;

ERROR: *.AFM is bad.

remedy: Not available. The AFM file contains an error and cannot be processed by addpsfnt.

message: ERROR: *font* is already in C:\EMTEX\PS\PSFONTS.MAP

remedy: The font has already been installed. If you want to install it nevertheless, you have to follow the instructions listed in Appendix E under error message 127 before calling addpsfnt again.

CHAPTER 5 — Footnotes, endnotes and filenotes

This chapter explains how to enter footnotes, endnotes and filenotes and how to change the appearance of these notes. Footnotes are printed in the critical apparatus on the same page, which contains the corresponding lemma the footnote refers to. Endnotes can be printed at any position in the text. You can instruct CET to print the endnotes at the current position in the text by inserting the corresponding command into the text (e. g., at the end of a chapter or at the end of your edition). Filenotes are similar to footnotes, but whereas footnotes are printed in the critical apparatus, filenotes are stored in a file on your hard disk (possible use: creation of a draft of an index of authors).

CET manages up to nine independent series of footnotes, endnotes and filenotes.

Footnotes, endnotes and filenotes are valid within numbered text only (see 2.5 Line numbering).

1. Footnotes

You can adjust each of the nine footnote series to your needs independently of the other footnote series. In the following text * must be replaced with a number between 1 and 9. The commands refer to the footnote series defined by this number.

The width and the height of the horizontal rule above a footnote series can be defined with FOOTNOTE*_RULE (2.0 in 0.4 pt, *value value*). The first value defines the width, the second value defines the height of the rule. You can suppress printing of the rule by specifying a width of 0 pt.

The formatting of the whole footnote series in the critical apparatus can be defined with FOOTNOTE*_FORMAT (normal *paragraph twocol threecol, *selection*). All four options for formatting a footnote series are demonstrated in the critical apparatus on the current page (unjustified print, justified print, two-column print and three-column print).

1 lemma 1] variant 1
1 lemma 2] variant 2
1 lemma 3] variant 3

1 lemma 1] variant 1 1 lemma 2] variant 2 1 lemma 3] variant 3 1 lemma 4] variant 4 1 lemma 5] variant 5 1 lemma 6] variant 6 1 lemma 7] variant 7 1 lemma 8] variant 8 1 lemma 9] variant 9

1 lemma 1] variant 1
1 lemma 2] variant 2
1 lemma 3] variant 3

1 lemma 4] variant 4
1 lemma 5] variant 5
1 lemma 6] variant 6

1 lemma 1] variant 1
1 lemma 2] variant 2
1 lemma 3] variant 3

1 lemma 4] variant 4
1 lemma 5] variant 5
1 lemma 6] variant 6

1 lemma 7] variant 7
1 lemma 8] variant 8
1 lemma 9] variant 9

A typical footnote looks like the following example:

1 et] *om.* A

The footnote consists of a line number (1), a lemma (et), a separator (]) and a variant reading (*om.* A). In order to create this footnote you have to enter the following text:

```
simpliciter (*et*) {<IB>om.<IE> A} purum
```

The lemma must be enclosed in (*...*). In addition, the commands enclosing the lemma determine the footnote series the footnote is put into. The commands can be defined with FOOTNOTE*_BEGIN/END:

FOOTNOTE1_BEGIN/END	(* *)
FOOTNOTE2_BEGIN/END	(+ +)
FOOTNOTE3_BEGIN/END	(- -)
FOOTNOTE4_BEGIN/END	(= =)
FOOTNOTE5_BEGIN/END	(# #)
FOOTNOTE6_BEGIN/END	(\$ \$)
FOOTNOTE7_BEGIN/END	(& &)
FOOTNOTE8_BEGIN/END	(^ ^)
FOOTNOTE9_BEGIN/END	{ }

The variant reading must be enclosed in {...}. The commands enclosing the variant reading can be defined with FOOTNOTE*_VAR_BEGIN/END (default for all nine footnote series: { }).

If the lemma in the text must be replaced with a different text to be printed in the critical apparatus (e. g., in order to abbreviate a lemma — however, CET can abbreviate lemmata for you) you can supply a so-called alternative lemma within {-...-} between the actual lemma and the corresponding variant reading:

```
simpliciter (*et*) {-ET-} {<IB>om.<IE> A} purum
```

FOOTNOTE*_A_L_BEGIN/END define the commands for marking alternative lemmata (default for all nine footnote series: {- -}).

The font for the line numbers in front of the lemmata in the critical apparatus can be defined with FOOTNOTE*_LINE_NUMBER_FONT (Times m m n 10.0 pt 10.0 pt, *font*). FOOTNOTE*_LINE_NUMBER_OMIT (*N Y, *yes/no*) determines whether the the line numbers in front of the lemmata in the critical apparatus are omitted (*Y) or not (*N).

The font for the lemmata in the critical apparatus can be defined with FOOTNOTE*_LEMMATA_FONT (- - - 10.0 pt 10.0 pt, *font*). FOOTNOTE*_LEMMATA_OMIT (*N Y,

1 et] *om.* A 1 ET] *om.* A

yes/no) determines whether the lemmata in the critical apparatus are omitted (*Y; useful for a footnote series containing references) or not (*N).

If FOOTNOTE*_LEMMATA_ABBREVIATE (N *Y, *yes/no*) is active (*Y), CET abbreviates lemmata consisting of more than two words. If FOOTNOTE*_LEMMATA_LOWER_CASE (*N Y, *yes/no*) is active (*Y), CET prints all lemmata in critical apparatus using lower case letters).

The separator between the lemma and the variant reading printed in the critical apparatus can be defined with FOOTNOTE*_SEPARATOR ([, *text*), the corresponding font can be defined with FOOTNOTE*_SEPARATOR_FONT (Times m m n 10.0 pt 10.0 pt, *font*).

The font for the variant readings in the critical apparatus can be defined with FOOTNOTE*_VARIANT_FONT (- - - 10.0 pt 10.0 pt, *font*).

Using FOOTNOTE*_LINE_NUMBER_REPEAT (N *Y, *yes/no*) you can instruct CET to repeat the line number in front of the lemmata in the critical apparatus, even if the lemmata refer to the same line in the text (*Y). Using FOOTNOTE*_LINE_NUMBER_REPEAT *N you can instruct CET to put the line number in front of the first lemma within a specific line in the text only and to separate all following lemmata in the same line by printing a lemma separator between two lemmata in the critical apparatus. The lemma separator can be defined with FOOTNOTE*_LEMMATA_SEPARATOR (||, *text*). The font for the lemma separator can be defined with FOOTNOTE*_LEMMATA_SEPARATOR_FONT (Times m m n 10.0 pt 10.0 pt, *font*).

CET can print a user defined text at the beginning of a footnote series. You must switch on this feature by specifying *Y after FOOTNOTE*_SIGLA (*N Y, *yes/no*). The text to be printed at the beginning of a footnote series must be specified between <S*B>...<S*E> in the main text. The font for this text can be defined with FOOTNOTE*_SIGLA_FONT (Times m m n 10.0 pt 10.0 pt, *font*) and the horizontal distance between this text and the actual footnotes can be defined with FOOTNOTE*_SIGLA_DISTANCE (4.0 em, *value*).

FOOTNOTE*_SIGL_BEGIN (<S*B>, *command name*)
 FOOTNOTE*_SIGL_END (<S*E>, *command name*)

Ordinary footnotes, which are connected by a number or a special character to the place in the text they refer to, can be obtained using the following settings:

FOOTNOTE*_FORMAT *normal paragraph twocol threecol
 FOOTNOTE*_LINE_NUMBER_OMIT N *Y
 FOOTNOTE*_LEMMATA_OMIT *N Y
 FOOTNOTE*_SEPARATOR
 FOOTNOTE*_LINE_NUMBER_REPEAT N *Y

Example:

```
lemma(<RB>a<RE>{) {variant reading}
```

An example of overlapping and nested footnotes follows:

```
(*simpliciter (*et*) {<IB>om.<IE> A}*) {<IB>om.<IE> B} purum
```

Using the commands discussed so far you can format all lemmata with two exceptions: lemmata, which cover several paragraphs, and overlapping but unnested lemmata

Footnotes covering several paragraphs must be formatted using the following method: The beginning of the lemma must be marked with a label — e. g., lembeg — (see 6. References). After the end of the lemma you must specify an empty lemma along with an alternative lemma consisting of the first and the last word of the lemma covering several paragraphs: (**) {-#lembeg beginning ... end-} {variant reading} (you have to change (**) according to the corresponding footnote series). Avoid spaces in front of (**), which would cause an additional space in the printout. Thus, in the case of a lemma covering several paragraphs you have to specify an alternative lemma.

Example:

```
-----
The @lembeg first paragraph.
```

```
The second(**) {-#lembeg first ... second-} {variant reading} paragraph.
-----
```

An example of overlapping but unnested footnotes:

```
This (*is an (*^#example of*) {<IB>om.<IE> C} overlapping
but unnested footnotes#^*) {<IB>om.<IE> D}.
```

The symbol ^ is fixed by CET, whereas the symbol # may be any character. By putting ^# right after the command name for the beginning of the lemma and by putting #^ right in front of the command name for the end of the lemma in the example above, you instruct CET to attach the first (* to the first *) and to attach the second (* to the second *). Without this special formatting you would get the following printout:
1–2 is ... footnotes] om. D and 1 example of] om. C

1 simpliciter et] om. B 1 et] om. A 1–2 first ... second] variant reading 1 is ... of] om. C
1–2 example ... footnotes] om. D

^a variant reading

Exercise 10:

Use footnote series 1 for the sigla of your manuscripts and for the folio numbers. Use footnote series 2 for the variant readings and footnote series 3 for the references.

footnote series 1: Put the sigla AD'J'K'P'R'S'T'W' at the beginning of the series. Suppress the separators (|) and the lemmata in the critical apparatus.

footnote series 3: Suppress the separators (|) and the lemmata in the critical apparatus.

Mark the following places in the text with | and put the following texts, which belong to the corresponding |, into footnote series 1:

Circa primum	Bad. II f. 3 ^{vZ}
Primo sic.	Bad. II f. 2 ^{rH}
Omne perfectum est	K' 284 ^{rb}
stat in eo	P' 233 ^{rb}

Put the following variant readings into footnote series 2 (the texts in brackets are provided for better orientation only):

ARTICULUS ... PERFECTIONE] XLII^{us} articulus de perfectione Dei D' om. AJ'K'P'R'S'T'W'
 (Sequitur de) Dei perfectione] inv. D'
 (Circa) quam] quod D'
 (inquirenda:) primo] primum D'J'K'P'S'T'W'
 (perfectus;) secundo] secundum D'J'K'P'R'S'T'W'
 (secundo,) si] de S'
 Utrum ... perfectus] om. AD'J'K'P'R'S'T'W'
 (non est) factus] effectus K'P'
 (per ipsum) facta] ita S'
 sermone I^o] om. D'J'K'P'R'S'T'W'
 (Secundo sic.) Id] illud K'
 (quod) recipit] recipiat T'
 (purum,) ut] Deus add. S'
 (extra) illa] om. S'

Put the following references into footnote series 3 (the texts in brackets are the lemmata, which are omitted in the critical apparatus):

(Deus ... I o a n n e m) AUGUST., *In Ioannis Evangelium*, tract. 1, n. 12 (CC lat. 36, p. 7,11, 14; PL 35, 1385); cf. Sermo 118, n. 1 (PL 38, 672): «Si autem omnia per ipsum facta sunt, intellige, quia non est factus ipse»; cf. *Ioann.*, I, 3; cf. THOM. DE AQ., *Summa theol.*, Ia, q. 4, a. 1 ad 1: «Quod enim factum non est, perfectum proprie dici non potest ... ».

(ut iam dicetur) Cf. *infra*, p. 23,130–133.

(habitum est supra) HENR. DE GAND., *Quaest. ord. (Summa)*, art. 21, q. 4 (ed. 1520 I, f. 228r-vX), q. 5 (ibid., f. 229rD).

(ut sunt ... huiusmodi.) Cf. HENR. DE GAND., *Quaest. ord. (Summa)*, art. 32, q. 1 (ed. R. MACKEN, p. 35–52).

(ut dicit PHILOSOPHUS) ARIST., *Metaph.*, V, c. 16 (ed. R. PONZALLI, p. 178,2–3; Iunt. VIII, f. 130 rD; 1021b 12–13); cf. ANON., *Auct. Arist.* (ed. J. HAMESSE, 1, 138).

2. Endnotes

You can adjust each of the nine endnote series to your needs independently of the other endnote series. In the following text * must be replaced with a number between 1 and 9. The commands refer to the endnote series defined by this number.

In contrast to footnotes you must instruct CET to print the endnotes. You can instruct CET to print all endnote series at the current position in the text using #E. You can instruct CET to print a specific endnote series at the current position using #E*.

```
ENDNOTES_PRINT           (#E, command name)
ENDNOTE*_PRINT          (#E*, command name)
```

A typical endnote differs from a typical footnote only in the command names enclosing the lemma:

```
simpliciter [*et*] {<IB>om.<IE> A} purum
```

The lemma must be enclosed in [*...*]. In addition, the commands enclosing the lemma determine the endnote series the endnote is put into. The commands can be defined with ENDNOTE*_BEGIN/END:

```
ENDNOTE1_BEGIN/END      [* *]
ENDNOTE2_BEGIN/END      [+ +]
ENDNOTE3_BEGIN/END      [| |]
ENDNOTE4_BEGIN/END      [= =]
ENDNOTE5_BEGIN/END      [# #]
ENDNOTE6_BEGIN/END      [$ $]
ENDNOTE7_BEGIN/END      [& &]
ENDNOTE8_BEGIN/END      [^ ^]
ENDNOTE9_BEGIN/END      [} {]
```

In analogy with the footnotes the command names for the alternative lemmata can be defined with ENDNOTE*_A_L_BEGIN/END (default for all nine endnote series: {- -}), the command names for the variant readings can be defined with ENDNOTE*_VAR_BEGIN/END (default for all nine endnote series: { }).

The following keywords define the appearance of the endnotes in analogy with the corresponding keywords for footnotes (see 5.1 Footnotes):

```
ENDNOTE*_LINE_NUMBER_FONT      Times m m n 10.0 pt 10.0 pt
ENDNOTE*_LINE_NUMBER_OMIT      *N Y
ENDNOTE*_LEMMA_FONT            - - - - 10.0 pt 10.0 pt
ENDNOTE*_LEMMA_OMIT            *N Y
ENDNOTE*_LEMMA_ABBREVIATE      N *Y
```

ENDNOTE*_LEMMA_LOWER_CASE	*N Y
ENDNOTE*_SEPARATOR_FONT	Times m m n 10.0 pt 10.0 pt
ENDNOTE*_SEPARATOR]
ENDNOTE*_VARIANT_FONT	- - - - 10.0 pt 10.0 pt

The notes concerning the special cases (lemmata covering several paragraphs and overlapping but unnested lemmata; see 5.1 Footnotes) apply to endnotes accordingly.

1 et] *om. A*

3. Filenotes

You can adjust each of the nine filenote series to your needs independently of the other filenote series. In the following text * must be replaced with a number between 1 and 9. The commands refer to the filenote series defined by this number.

Filenotes consist of text, which CET puts into a file along with the corresponding position of the filenote within the text (i. e., page number, line number and subline number). E. g., using filenotes you can create an index of authors: provide each work quoted in the text with a filenote containing its author, the name of the work and the corresponding passage within the work. CET generates a file which contains all these filenotes. This file may serve as a draft of an index of authors.

A typical filenote looks like the following example:

```
<1 Arist., Metaph., V, c. 16 1>
```

Using default values CET puts the filenote above into the file FILENOTE.1 as follows:

```
Arist., Metaph., V, c. 16 38, 1
```

38 is the number of the page containing the filenote and 1 is the number of the line containing the filenote.

The text of the filenote must be enclosed in <1...1>. In addition, the commands enclosing the text of the filenote determine the filenote series the filenote is put into. The commands can be defined with FILENOTE*_BEGIN/END:

FILENOTE1_BEGIN/END	<1 1>
FILENOTE2_BEGIN/END	<2 2>
FILENOTE3_BEGIN/END	<3 3>
FILENOTE4_BEGIN/END	<4 4>
FILENOTE5_BEGIN/END	<5 5>
FILENOTE6_BEGIN/END	<6 6>
FILENOTE7_BEGIN/END	<7 7>
FILENOTE8_BEGIN/END	<8 8>
FILENOTE9_BEGIN/END	<9 9>

You can define the name of the file for a filenote series with FILENOTE*_FILENAME (FILENOTE.*, *filename*).

FILENOTE*_TEXT_POSITION (*left right, *selection*) determines whether the text of the filenote (left) or the position of the filenote within the text (right) is put into the file first. If you are going to sort your filenote file using an external program, you should select left.

<PNB>18<PNE>

<S1B>AD'J'K'P'R'S'T'W'<S1E>

<LAT>

#N+ #L+

<[- - - 16 pt 18 pt]>

<CB>ARTICULUS @M1 XLII<CE>

<CB>DE DEI PERFECTIIONE(++) {-#M1 ARTICULUS ... PERFECTIIONE-} {XLII<RB>us<RE> articulus de perfectione Dei D' <IB>om.<IE> AJ'K'P'R'S'T'W'}<CE>

<VSB>0.7 cm<VSE>

<[- - - 12 pt 14 pt]>

Sequitur de (+Dei perfectione+) {<IB>inv.<IE> D'}. Circa (+quam+) {quod D'} duo sunt inquirenda: (+primo+) {primum D'J'K'P'R'S'T'W'}, si Deus possit dici perfectus; (+secundo+) {secundum D'J'K'P'R'S'T'W'}, (+si+) {de S'} perfectione cuiuslibet creaturae sit perfectus.

#L- #N-

#N+ #L+

<VSB>0.4 cm<VSE>

<CB>QUAESTIO 1<CE>

<VSB>0.4 cm<VSE>

<CB><SCB> (+Utrum Deus possit dici perfectus+) {<IB>om.<IE> AD'J'K'P'R'S'T'W'}<SCE><CE>

<VSB>0.4 cm<VSE>

(*|*) {Bad. II f. 3<RB>vZ<RE>} Circa primum arguitur, quod Deus non possit dici perfectus.

Primo (*|*) {Bad. II f. 2<RB>rH<RE>} sic. Omne perfectum (*|*) {K' 284<RB>rb<RE>} est factum, quia praepositio illa 'per' in proposito non diminuit, sed potius auget. (-Deus <<<IB>non est (+factus+) {effectus K'P'}, quia per ipsum (+facta+) {ita S'} sunt omnia.<IE>>>, ut vult <SCB>Augustinus<SCE>, (+sermone I<RB>o<RE>+) {<IB>om.<IE> D'J'K'P'R'S'T'W'} <SPB>Super Ioannem.<SPE>-) {<SCB>August.<SCE>, <IB>In Ioannis Evangelium<IE>, tract. 1, n. 12 (CC lat. 36, p. 7,11, 14; PL 35, 1385); cf. Sermo 118, n. 1 (PL 38, 672): <<Si autem omnia per ipsum facta sunt, intellige, quia non est factus ipse>>; cf. <IB>Ioann.<IE>, I, 3; cf. <SCB>Thom. de Aq.<SCE>, <IB>Summa theol.<IE>, Ia, q. 4, a. 1 ad 1: <<Quod enim factum non est, perfectum proprie dici non potest ...>>.) Ergo etc.

Secundo sic. (+Id+) {illud K'}, cuius natura stat (*|*) {P' 233<RB>rb<RE>} in eo quod supra se recipit omnes conditiones nobilitatis, maxime imperfectum est, quia de ratione perfecti est quod non recipit additionem in dignitate et nobilitate, et de ratione imperfecti quod (+recipit+) {recipiat T'} additionem, (-ut iam dicitur.-) {Cf. <IB>infra<IE>, p. 23,130--133.} Deus est huiusmodi, quia non est nisi esse purum, (+ut+) {Deus <IB>add.<IE> S'} (-habitus est supra.-) {<SCB>Henr. de Gand.<SCE>, <IB>Quaest. ord. (Summa)<IE>, art. 21, q. 4 (ed. 1520 I, f. 228r-vX), q. 5 (ibid. f. 229rD).} Esse autem simpliciter et purum super se recipit omnes determinationes dignitatis et nobilitatis, (-ut sunt sapientia, bonitas pulchritudo, et cetera huiusmodi.-) {Cf. <SCB>Henr. de Gand.<SCE>, <IB>Quaest. ord. (Summa)<IE>, art. 32, q. 1 (ed. <SCB>R. Macken<SCE>, p. 35--52).} Ergo etc.

Tertio sic. Quae perfecta sunt per se, nihil invenitur extra (+illa+) {<IB>om.<IE> S'}, (-ut dicit <SCB>Philosophus<SCE>-) {<SCB>Arist.<SCE>, <IB>Metaph.<IE>, V, c. 16 (ed. <SCB>R. Ponzalli<SCE>, p. 178,2--3; Iunt. VIII, f. 130 rD; 1021b 12--13); cf. <SCB>Anon.<SCE>, <IB>Auct. Arist.<IE> (ed. <SCB>J. Hamesse<SCE>, 1, 138).}, V<RB>o<RE> <SPB>Metaphysicae,<SPE> cap. De perfecto. Et dicit, I<RB>o<RE> <SCB>Caeli et mundi,<SCE> quod corpus universi ex hoc est perfectum, quia non habet aliud corpus extra, quod tangit ipsum. Extra Deum autem sunt omnes creaturae. Ergo etc.

#L- #N-

ARTICULUS XLII
DE DEI PERFECTIOE

Sequitur de Dei perfectione. Circa quam duo sunt inquirenda: primo, si Deus possit dici perfectus; secundo, si perfectione cuiuslibet creaturae sit perfectus.

QUAESTIO 1

UTRUM DEUS POSSIT DICI PERFECTUS

| Circa primum arguitur, quod Deus non possit dici perfectus.
Primo | sic. Omne perfectum | est factum, quia praepositio illa 'per' in proposito non diminuit, sed potius auget. Deus «*non est factus, quia per ipsum facta sunt omnia.*», ut vult AUGUSTINUS, sermone I^o Super Ioannem. Ergo etc.
Secundo sic. Id, cuius natura stat | in eo quod supra se recipit omnes conditiones nobilitatis, maxime imperfectum est, quia de ratione perfecti est quod non recipit additionem in dignitate et nobilitate, et de ratione imperfecti quod recipit additionem, ut iam dicetur. Deus est huiusmodi, quia non est nisi esse purum, ut habitum est supra. Esse autem simpliciter et purum super se recipit omnes determinationes dignitatis et nobilitatis, ut sunt sapientia, bonitas pulchritudo, et cetera huiusmodi. Ergo etc.
Tertio sic. Quae perfecta sunt per se, nihil invenitur extra illa, ut dicit PHILOSOPHUS, V^o Metaphysicae, cap. De perfecto. Et dicit, I^o Caeli et mundi, quod corpus universi ex hoc est perfectum, quia non habet aliud corpus extra, quod tangit ipsum. Extra Deum autem sunt omnes creaturae. Ergo etc.

AD'J'K'P'R'S'T'W' 3 Bad. II f. 3^{vz} 4 Bad. II f. 2th 4 K' 284^b 8 P' 233^b
1–2 ARTICULUS ... PERFECTIOE] XLII^{us} articulus de perfectione Dei D' om. AJ'K'-
P'R'S'T'W' 3 Dei perfectione] im. D' 3 quam] quod D' 3 primo] primum D'J'K'-
P'S'T'W' 4 secundo] secundum D'J'K'P'R'S'T'W' 4 si] de S' 2 UTRUM ... PER-
FECTUS] om. AD'J'K'P'R'S'T'W' 5 factus] effectus K'P' 6 facta] ita S' 6 sermone
I^o] om. D'J'K'P'R'S'T'W' 8 Id] illud K' 11 recipit] recipiat T' 12 ut] Deus add.
S' 15 illa] om. S'

5–7 AUGUST., *In Ioannis Evangelium*, tract. 1, n. 12 (CC lat. 36, p. 7,11, 14; PL 35, 1385);
cf. Sermo 118, n. 1 (PL 38, 672): «Si autem omnia per ipsum facta sunt, intellige, quia non
est factus ipse»; cf. *Ioann.*, I, 3; cf. THOM. DE AQ., *Summa theol.*, Ia, q. 4, a. 1 ad 1: «Quod
enim factum non est, perfectum proprie dici non potest ... ». 11 Cf. *infra*, p. 23,130–133.
12 HENR. DE GAND., *Quaest. ord. (Summa)*, art. 21, q. 4 (ed. 1520 I, f. 228r-vX), q. 5
(ibid. f. 229rD). 13–14 Cf. HENR. DE GAND., *Quaest. ord. (Summa)*, art. 32, q. 1 (ed.
R. MACKEN, p. 35–52). 15–16 ARIST., *Metaph.*, V, c. 16 (ed. R. PONZALLI, p. 178,2–3;
Iunt. VIII, f. 130 rD; 1021b 12–13); cf. ANON., *Auct. Arist.* (ed. J. HAMESSE, I, 138).

CHAPTER 6 — References

Using the reference commands you can put symbolic references into your text. These symbolic references are replaced by CET with the corresponding page and line numbers. There are two kinds of references within CET: simple references and from-to references. A simple reference points to specific place (one word) within the text, whereas a from-to reference points to a specific passage (several words) within the text. The counterpart of a reference is a label, a symbol a reference refers to.

A label must be enclosed in the command names defined with REF_LABEL_BEGIN/END (`@/ , command name`). As the default for REF_LABEL_END is empty, under default conditions a label consists of `@` and the label name. The only valid characters within a label name are the letters A...Z, a...z and the figures 0...9. Examples for labels: `@label1` and `@a44a1` and `@L2`.

You refer to a label with `@(label name)`. CET replaces this symbolic reference with the text defined with REF_SIMPLE_TEXT (p. %p,%l.%s, *text*) or REF_SIMPLE_TEXT_SAME_PAGE (l. %l.%s, *text*). %p, %l and %s within these texts are templates and are replaced with the page number, line number and subline number of the corresponding label. If the label and the reference are on the same page, the reference is replaced with the text defined with REF_SIMPLE_TEXT_SAME_PAGE, otherwise the text is replaced with the text defined with REF_SIMPLE_TEXT. Thus a typical reference has the following form: p. 1,1.1

REF_SIMPLE_BEGIN	(@(<i>, command name</i>)
REF_SIMPLE_END	(<i>, command name</i>)
REF_SIMPLE_TEXT	(p. %p,%l.%s, <i>text</i>)
REF_SIMPLE_TEXT_SAME_PAGE	(l. %l.%s, <i>text</i>)

If you use a from-to reference, you have to supply two labels. The first label marks the beginning and the second label marks the end of the passage, which the reference refers to. A from-to reference consists of the command names enclosing the from-to reference (`@[...]`) and a separator (`,`), which separates the two labels within the from-to reference.

REF_DOUBLE_BEGIN	(@[<i>, command name</i>)
REF_DOUBLE_SEPARATOR	(<i>, , command name</i>)
REF_DOUBLE_END	(<i>], command name</i>)

Using the defaults a from-to reference has the following form:

```
@[label1, label2]
```

CET replaces a from-to reference with the page numbers, line numbers and subline numbers corresponding to the two labels, which denote the beginning and the end of the passage the from-to reference refers to. The format of the printed reference is defined by the following keywords (a space must be entered as `_`):

```
REF_DOUBLE_TEXT           (p._, text)
REF_DOUBLE_TEXT_SAME_PAGE (l._, text)
REF_DOUBLE_TEXT_AFTER_PAGE (, , text)
REF_DOUBLE_TEXT_AFTER_LINE (., text)
REF_DOUBLE_TEXT_SEPARATOR (-, text)
```

Thus a typical from-to reference has the following form: p. 1,1.1–2,2.1

If both labels within a from-to reference and the reference itself are on the same page, a typical from-to reference has the following form: l. 1.1–2.2

If both labels within a from-to reference and the reference itself are on the same page and you do not want CET to omit the page number, you can turn off the omission of the page number with `REF_DOUBLE_TEXT_OMIT_SAME_PAGE *N`.

CET abbreviates from-to references. E. g., CET replaces the reference p. 3,1–3,23 with the reference p. 3,1–23. You can turn off the abbreviation of references with `REF_DOUBLE_TEXT_ABBREVIATE *N`.

Examples:

input	with default settings	REF_DOUBLE_TEXT_OMIT_SAME_PAGE *N
This @label1 word: @(label1).	This word: l. 1.	This word: p. 1,1.
This @label1 is a demonstration of a from-to reference to a passage which covers several lines.@label2 The position of the first sentence is: @[label1,label2]	This is a demonstration of a from-to reference to a passage which covers several lines. The position of the first sentence is: l. 1–3	This is a demonstration of a from-to reference to a passage which covers several lines. The position of the first sentence is: p. 1,1–3
input	with default settings	REF_DOUBLE_TEXT_ABBREVIATE *N
This @label1 is a @label2 demonstration: @[label1,label2]	This is a demonstration: l. 1	This is a demonstration: l. 1–1

CHAPTER 7 — Indices

There are four kinds of indices within CET: the index of words from the main text, the index of words from the variant readings, the index of labels and the index of references. The index of words from the main text contains the places of all words from the main text (all words except words from variant readings) you requested CET to index. The index of words from the variant readings contains the places of all words from from the variant readings you requested CET to index. The index of labels contains all labels defined within your edition and the index of references contains all references used within your edition.

1. Index of words from the main text

You supply a list of words and CET generates a list containing all places of these words in the main text (i. e., all words except words from variant readings).

The list of words you supply must be put into a file with the same name as the file to be typeset, but with the filename extension `.IW`. After calling Typeset the index of words from the main text can be found in the file `*.IWF`

The following rules apply to the file `*.IW` containing the list of words:

- only one word per line
- leading and trailing spaces are ignored
- empty lines are ignored

The following rules apply to the search for words listed in the file `*.IW`:

The search is case-insensitive.

A word ending with `*` matches all words beginning with the same word without the `*` (e. g., `et*` matches the words `et`, `etiam`, `etsi`, ...).

If the file `*.IW` contains the words `et*` and `etiam`, `etiam` is listed under `et*`, but not under `etiam`; if you need a list for `et*` *and* `etiam`, you have to call Typeset twice: Create a file `*.IW` containing `et*` and call Typeset. Rename the file `*.IWF`. Remove `et*` from the file `*.IW`, put `etiam` into it and call Typeset again.

Only words coinciding literally are listed in the file `*.IWF`:
`scilicet` does not match `sci(*|*){...}licet`.

Multiple occurrences of the same word in the same line are indicated by the corresponding number of occurrences within brackets.

Example:

Text to be typeset:

```
----- DEMO.TXT -----
This is a test of indices within CET. CET is
an abbreviation for 'Critical Edition Typesetter', for a
program for typesetting critical editions.
----- DEMO.TXT -----
```

List of words to be indexed:

```
----- DEMO.IW -----
is
a*
this
for
----- DEMO.IW -----
```

Index of words from the main text generated by CET:

```
----- DEMO.IWF -----
a* 1,1; 1,2 (3)

for 1,2 (2); 1,3

is 1,1 (2)

this 1,1
----- DEMO.IWF -----
```

Exercise 11:

Create an index of words from the main text for the following words:
et, ut, Deus

2. Index of words from the variant readings

You supply a list of words and CET generates a list containing all places of these words in the variant readings. You can use this index for creating an index containing all lemmata with a specific combination of sigla.

The list of words must be put into a file with the same name as the file to be typeset, but with the filename extension .IM. After calling Typeset the index of words from the variant readings can be found in the file *.IMF. The file *.IMT contains the complete text of the variant readings and the text of the corresponding lemmata. The rules for the format of the file *.IM, the rules for the search for the words from the list of words and the rules for the format of the files *.IMF and *.IMT correspond with the rules in the previous section (see 7.1 Index of words from the main text).

Example:

Text to be typeset:

```

----- DEMO.TXT -----
This (*is*) {om. AK} a (*test*) {om. AKP} of indices
within (*CET*) {author: Bernt Karasch et al.}.
CET (*is*) {om. AK} an abbreviation for 'Critical Edition
Typesetter', for (*a*) {om. AKP} program for typesetting
(*critical*) {om. AK} editions.
----- DEMO.TXT -----

```

List of words to be indexed:

```

----- DEMO.IM -----
AKP
AK
Bernt
----- DEMO.IM -----

```

Index files generated by CET:

```

----- DEMO.IMF -----
AK 1,1; 1,3; 1,5

AKP 1,1; 1,4

Bernt 1,2
----- DEMO.IMF -----

```

and

```

----- DEMO.IMT -----
AK

1, 1      is ] om. AK
1, 3      is ] om. AK
1, 5      critical ] om. AK

AKP

1, 1      test ] om. AKP
1, 4      a ] om. AKP

Bernt

1, 2      CET ] author: Bernt Karasch et al.
----- DEMO.IMT -----

```

Exercise 12:

Create an index of words from the variant readings for the following words:
S', D', cf

3. Index of labels

After calling Typeset the file *.LBL contains a list of all labels and their positions within the text.

Example:

Label	Defined on page
one	1, 23
two	1, 24
xone	1, 25. 1
xtwo	1, 25. 2

Label names must be unique. Should you have defined a specific label more than once by mistake, you can use the index of labels for discovering such mistakes.

4. Index of references

After calling Typeset the file *.REF contains a list of all references and their positions within the text.

Example:

Reference	Used on page	Resolved
@[one,two]	1, 1	1, 23 - 1, 24
@(one)	* 1, 2	1, 23
@[xone,xtwo]	1, 25. 3	1, 25. 1 - 1, 25. 2
@(one)	1, 25. 4	1, 23

An asterisk (*) in front of the position indicates that the reference is in the critical apparatus (i. e. within an alternative lemma or within a variant reading). In this case the position listed under 'Used on page' is the beginning of the corresponding lemma, as the critical apparatus has no line numbers.

ERROR at the end of a line indicates that the reference is outside of numbered text or the reference refers to non-existent labels.

CHAPTER 8 — Creating an edition — step by step

1. Create a new text file (using Edit) and call Typeset in order create the corresponding default configuration file. For more complex editions you should put the sections of your edition into separate text files and include these text files from the main file with <IFB>...<IFE> (see 3.8 Including text files).
2. Call Settings and adapt the page layout and the line numbering to your needs (see Chapter 2).
3. If the fonts you are going to use within your editions do not belong to the default fonts provided by CET, you have to install your fonts (see Chapter 4).
4. Enter the text of your edition along with the necessary CET commands (see Chapter 3, 5, 6). As most CET commands consist of a pair of commands, it is recommended to enter a pair of CET commands first and to enter the text between the CET commands afterwards. By using this method you can avoid forgetting required CET commands. This method is especially useful for long lemmata and variant readings, because the danger of forgetting a closing bracket is especially high due to the length of the insertions.
5. If necessary, create an index of words from the main text and/or an index of words from the variant readings. Make sure that you have not defined a specific label more than once (check the label index). Check whether the reference index contains lines containing the word ERROR and correct these errors (see Chapter 7).
6. Configure CET for the resolution of the typesetter used in the printing office (see \CET\INSTALL.ENG, 5. Page 6 of 8, Changing the typesetter resolution after the installation).
7. Typeset and print your edition. If you are satisfied with the printout, copy the PostScript file (*.PS) to a diskette and send the diskette to the printing office.

APPENDIX

A. Limitations

Please note the following limitations:

During the translation into a \TeX file a paragraph must not contain more than 65535 characters (this corresponds to about 64000 characters before the translation).

If you are using BOXER and if you change the default command names you have to change `\BOXER\DEFAULT.CFG` after the line "EXT=.TXT,.CFG,.ERR" accordingly, so that the changed command names are highlighted within BOXER (see `\BOXER\BOXER.DOC`, Chapter 21).

The following label names are reserved and must not be used (<number> is an integer number (see Appendix B, Number)):

f<number>	(used by CET for filenotes)
d<number>_<number>	(used by CET for marking lemmata)
i<number>_<number>	(used by CET for word indices)
m<number>	(used by CET for footnotes and endnotes)
n<number>	(used by CET for footnotes and endnotes)
r<number>	(used by CET for a list of references)

Text spaced out with <SPB>...<SPE> is hyphenated only at places marked with |-

Typesetting in two columns with <TCB>...<TCE> is under development. The following limitations apply:

Lemmata in the critical apparatus may refer to the following page (this problem might be solved by decreasing the number 1.7 in the line `\newdimen\tcvsize\tcvsize=1.7\vsz` in the file `\EMTEX\TEXINPUT\TWOCOL.TEX`). Crop marks are not supported. Text on the last page is not split into two columns of the same height.

The reference commands (`@(label)` and `@[label1,label2]`) ignore the settings specified with `CUT_LINE_NUMBER` and `CUT_SUBLINE_NUMBER`.

The following filename extensions are reserved and must not be used:

.AUX, .BAT, .CFG, .DVI, .END, .ERR, .FIP, .FIT, .FNT, .IM, .IMF, .IMS, .IMT, .IW, .IWF, .IWS, .LBL, .LBT, .LOG, .PS, .REF, .RET, .STY, .TEX

B. Parameter types

Value

A *value* consists of a number (an integer number or a number with decimal point and decimal fraction) and an unit, which is separated from the number by spaces. One of the following units must be specified:

abbreviation	name of the unit	conversions	
pt	point	1 pt = 0,0351 cm	1 pc = 12 pt
pc	pica	1 pc = 0,422 cm	
in	inch	1 in = 2,54 cm	
bp	big point	1 bp = 0,0353 cm	72 bp = 1 in
cm	centimetre		
mm	millimetre	1 mm = 0,1 cm	
dd	Didot point	1 dd = 0,0376 cm	
cc	Cicero	1 cc = 0,451 cm	1 cc = 12 dd
em	em	width of the M	
ex	ex	height of the x	

Example: 12 pt

Font

A font specification consists of six components, which are separated by spaces from each other:

1. Family:

AvantGarde
 Bookman
 Courier
 Helvetica
 NewCenturySchlbk
 Palatino
 Times
 ZapfChancery
 LevyGreek

and all names of PostScript fonts installed with addpsfnt (see 4. Adding fonts)

2. Weight:

abbreviation	name
ul	ultralight
el	extralight
l	light
sl	semilight
m	medium (normal)
sb	semibold
b	bold
eb	extrabold
ub	ultrabold

At the moment only m (medium) and b (bold) may be used.

3. Width:

abbreviation	name	
uc	ultracondensed	50 %
ec	extracondensed	62,5 %
c	condensed	75 %
sc	semicondensed	87,5 %
m	medium	100 %
sx	semiexpanded	112,5 %
x	expanded	125 %
ex	extraexpanded	150 %
ux	ultraexpanded	200 %

At the moment only m (medium) may be used.

4. Shape:

abbreviation	name
n	normal
it	italic
sl	slanted
sc	small capitals
u	upright italic

At the moment only n (normal), it (italic) and sc (small capitals) may be used. sl (slanted) is available only for PostScript fonts installed with addpsfnt (see 4. Adding fonts).

5. Size:

The parameter type for size is *value*. Recommended sizes are:
5 pt, 6 pt, ... , 25 pt

6. Leading:

The leading (the vertical distance between the base lines of two adjacent lines) is specified as *value*.

- instructs CET to use the current setting for the corresponding component. - may be used for the first to the fourth component only.

Examples:

```
Times m m it 12 pt 16 pt
```

This font definition instructs CET to use the font family Times with normal weight (first m), medium width (second m) and italic shape (it) with a size of 12 point (12 pt) and with a leading of 16 point (16 pt).

```
- - - - 16 pt 20 pt
```

This font definition instructs CET to set the size and the leading of the current font to 16 point and 20 point respectively.

Text

Text consists of any characters, which do not form CET command names.

Selection

A *selection* consists of several keywords separated from each other by spaces. Exactly one keyword must be marked with an asterisk (*).

Example: left right *inner outer

The option inner has been selected.

Yes/No

The parameter type *yes/no* can take the value yes or no. yes must be specified with N *Y (or *Y), no must be specified with *N Y (or *N). So the parameter type *yes/no* is a special case of a *selection*.

Number

A *number* is a positive integer number (1, 2, 3, ...).

Command name

A *command name* is a combination of characters you put into your text in order to instruct CET to do something special (e. g. printing with italic letters). All default command names may be changed in order to adapt CET to your needs. Unless stated otherwise, command names must not be empty, must not contain more than 10 characters and must be unique within the configuration file.

Space

A *space* has the following form:

value1 plus *value2* minus *value3*

CET tries to print a space according to *value1*, but may vary the space within the limits *value1 - value3* and *value1 + value2* in order to improve the line and page breaks.

Label name

A *label name* is a *text* containing only the letters A...Z, a...z and the figures 0...9.

Filename

A *filename* consists of an optional path, the name of the file and a filename extension separated from the name of the file by a period (see your MS-DOS user manual).

C. Solutions to the exercises

Exercise 1 (p. 6)

1. Type CET and press Return. The CET menu appears. Move the selection bar onto Edit using the cursor keys and press Return. Type SUMMA and press Return (using the keys ← and Del you can delete characters). CET displays the file SUMMA.TXT. Quit the word processing program (BOXER: Alt-X).
2. Move the selection bar onto Typeset and press Return. Accept the default (SUMMA) by pressing Return. Now CET typesets your text.
3. Move the selection bar onto Preview and press Return. Accept the default (SUMMA) and press Return. CET previews the text on the screen. Quit Preview by typing q. Proceed accordingly with PostScript-Preview (press Return or Ctrl-Pause to quit PostScript-Preview).
4. Call Print, accept SUMMA by pressing Return and press Return again in order to print the whole file SUMMA.TXT.
5. Call Settings and accept SUMMA by pressing Return. CET displays the configuration file SUMMA.CFG. Quit the word processing program (BOXER: Alt-X).

Exercise 2 (p. 12)

Call Settings and change the following lines as stated below:

HORIZONTAL_SIZE	130 mm
VERTICAL_SIZE	190 mm
GLOBAL_FONT	Times m m n 12.0 pt 14.0 pt
PARAGRAPH_INDENTATION	5 mm
WORD_GLUE	0.4 em plus 0.2 em minus 0.3 em

Quit the word processing program (BOXER: Alt-X, W), call Typeset and call Print. Center the printout on the paper by adjusting HORIZONTAL_OFFSET and VERTICAL_OFFSET (determine the shifts using a ruler, modify the settings and print the text again).

Exercise 3 (p. 13)

Call Settings and change the following lines as stated below:

CROP_MARK_WIDTH	0.5 pt
CROP_MARK_GAP	4.0 pt
CROP_MARK_HORIZONTAL_DISTANCE	16 cm
CROP_MARK_VERTICAL_DISTANCE	25 cm
CROP_MARK_HEAD_MARGIN	2 cm
CROP_MARK_BACK_MARGIN	2 cm

Quit the word processing program (BOXER: Alt-X, W) and call Typeset and Preview. Quit Preview (q).

Exercise 4 (p. 14)

Call Settings and change the following lines as stated below:

HEADLINE_VERTICAL_DISTANCE	0.75 cm
HEADLINE_POSITION	*inner center outer
HEADLINE_FONT	Times m m n 10.0 pt 12.0 pt
HEADLINE_TEXT_LEFT	ARTICULUS XLII
HEADLINE_TEXT_RIGHT	QUAESTIO 1
PAGE_NUMBER_FONT	Times m m n 12.0 pt 14.0 pt

Quit the word processing program (BOXER: Alt-X, W); call Edit. Put <PNB>18<PNE> at the beginning of the file, quit the word processing program (BOXER: Alt-X, W) and call Typeset and Preview. Quit Preview (q).

Exercise 5 (p. 15)

Call Edit. Put <LAT> at the beginning of the file, quit the word processing program (BOXER: Alt-X, W) and call Typeset and Preview. Quit Preview (q).

Exercise 6 (p. 17)

Call Settings and change the following lines as stated below:

LINE_NUMBER_FIRST	1
LINE_NUMBER_INCREMENT	1

Quit the word processing program (BOXER: Alt-X, W) and call Edit. Put #N+ #L+ at the beginning of the text, put #L- #N- and #N+ #L+ in front of QUAESTIO 1 and put #L- #N- after the whole text (each command in a paragraph of its own). Call Typeset and Preview. Quit Preview (q).

Exercise 7 (p. 20)

Add an empty line after the corresponding lines in order to start a new paragraph. Call Edit and use the following command names for formatting the text:

centered	<CB>...<CE>
small capitals	<SCB>...<SCE>
spaced out	<SPB>...<SPE>
italic	<IB>...<IE>
quotation marks	<<...>>
exponents	<RB>...<RE>

Quit the word processing program (BOXER: Alt-X, W) and call Typeset and Preview. Quit Preview (q).

Exercise 8 (p. 23)

Call Edit and insert <[- - - 16 pt 18 pt]> in front of the line ARTICULUS XLII in a paragraph of its own. Insert <[- - - 12 pt 14 pt]> after DE DEI PERFECTIO in a paragraph of its own. Quit the word processing program (BOXER: Alt-X, W) and call Typeset and Preview. Quit Preview (q).

Exercise 9 (p. 24)

Call Edit and insert <VSB>0.7 cm<VSE> and <VSB>0.4 cm<VSE> respectively at the corresponding places in a paragraph of its own. Quit the word processing program (BOXER: Alt-X, W) and call Typeset and Preview. Quit Preview (q).

Exercise 10 (p. 36)

Call Settings and change the following lines as stated below:

FOOTNOTE1_SIGLA	N *Y
FOOTNOTE1_SEPARATOR	
FOOTNOTE1_LEMMA_OMIT	N *Y
FOOTNOTE3_SEPARATOR	
FOOTNOTE3_LEMMA_OMIT	N *Y

Quit the word processing program (BOXER: Alt-X, W) and call Edit. The necessary commands can be found on page 40. Quit the word processing program (BOXER: Alt-X, W) and call Typeset and Preview. Quit Preview (q). Print the page by calling Print.

Exercise 11 (p. 45)

Create the following file:

```
----- SUMMA.IW -----
et
ut
Deus
----- SUMMA.IW -----
```

Call Typeset. Now the index of words from the main text can be found in the file SUMMA.IWF:

```
----- SUMMA.IWF -----
Deus 18,4; 18,2; 18,3; 18,5; 18,11

et 18,10 (2); 18,12; 18,13; 18,14; 18,16; 18,17

ut 18,6; 18,11; 18,12; 18,13; 18,15
----- SUMMA.IWF -----
```

Exercise 12 (p. 46)

Create the following file:

```
----- SUMMA.IM -----
S'
D'
cf
----- SUMMA.IM -----
```

Call Typeset. Now the index of words from the variant readings can be found in the files SUMMA.IMF and SUMMA.IMT:

```
----- SUMMA.IMF -----
cf 18,5 (3); 18,11; 18,13; 18,15

D' 18,2; 18,3 (2)

S' 18,4; 18,6; 18,12; 18,15
----- SUMMA.IMF -----
```

----- SUMMA. IMT -----

cf

- 18, 5 (3) Deus <<non est factus , quia per ipsum facta sunt omnia.>>, ut vult Augustinus, sermone Io Super Ioannem.]
 August., In Ioannis Evangelium, tract. 1, n. 12 (CC lat. 36, p. 7, 11, 14; PL 35, 1385); cf. Sermo 118, n. 1 (PL 38, 672): <<Si autem omnia per ipsum facta sunt, intellige, quia non est factus ipse>>; cf. Ioann., I, 3; cf. Thom. de Aq., Summa theol., Ia, q. 4, a. 1 ad 1: <<Quod enim factum non est, perfectum proprie dici non potest >>.
- 18, 11 ut iam dicetur.] Cf. infra, p. 23, 130--133.
- 18, 13 ut sunt sapientia, bonitas pulchritudo, et cetera huiusmodi.] Cf. Henr. de Gand., Quaest. ord. (Summa), art. 32, q. 1 (ed. R. Macken, p. 35--52).
- 18, 15 ut dicit Philosophus] Arist., Metaph., V, c. 16 (ed. R. Ponzalli, p. 178, 2--3; Iunt. VIII, f. 130 rD; 1021b 12--13); cf. Anon., Auct. Arist. (ed. J. Hamesse, 1, 138).

D'

- 18, 2] XLIIus articulus de perfectione Dei D' om.
 AJ'K'P'R'S'T'W'
- 18, 3 Dei perfectione] inv. D'
- 18, 3 quam] quod D'

S'

- 18, 4 si] de S'
- 18, 6 facta] ita S'
- 18, 12 ut] Deus add. S'
- 18, 15 illa] om. S'

----- SUMMA. IMT -----

D. Common problems and solutions

Problem: black rectangle at the end of a line

Solution: Black rectangles at the end of line indicate lines which are too wide (i. e., lines which are wider than the sum of the values defined with `HORIZONTAL_SIZE` and `HORIZONTAL_TOLERANCE`). \TeX was not able to find a good place for breaking the line. To avoid this situation you can use one or a combination of the following methods:

1. If there is no need to force \TeX to break the line, you can increase the value defined with `HORIZONTAL_TOLERANCE` or you can switch off the black rectangles with `MARK_BAD_LINES *N`.
2. Perhaps \TeX does not know how to hyphenate a word at the end of the line. Mark all possible places for a hyphen with `|-`.
3. Change the values for the spaces (`WORD_GLUE` for the main text, `APPARATUS_WORD_GLUE` for the critical apparatus and `FOOTNOTE_PARAGRAPH_GLUE` for the the distance between footnotes formatted in paragraphs): Increase the corresponding default values and the values defined with plus and minus.
4. Insert a horizontal space with `<HSB>value<HSE>` at a suitable place.

Problem: wrong line numbers in front of the lemmata in the critical apparatus

Solution: `WORD_DISTANCE` with a number different from 0 in combination with `LINEATION_BY *page` may cause wrong line numbers within the critical apparatus. Setting `BALLAST` to 100 should fix this problem. Note that complex texts may require several successive calls to `Typeset` before all line numbers are correct.

Problem: \TeX error message after `Typeset`

Solution: If \TeX stops with an error message and displays `?`: type `x` and press `Return`. Quit `CET`, type `CET CLEAR` and press `Return`. `Typeset` your text again.

Warning: `CET CLEAR` deletes all files, which `CET` can recreate from the text files and configuration files (i. e., `*.TEX`, `*.STY`, `*.PS`, `*.IWF`, ...).

E. Error messages

If CET detects an error in your text, CET displays an error message. This error message contains information about the type of the error, the name of the file in which the error occurred, the line number and the beginning of the text which caused the error.

Example:

```
----- SUMMA.ERR -----
Error   : 160 Matching command does not follow
Line 15 : <SCB>Augustinus, sermone Io Super Ioannem.
File    : SUMMA.TXT
----- SUMMA.ERR -----
```

When you cancel the error message by quitting the word processing program which displays the error message, CET loads the faulty text into the word processing program. If the word processing program supports a command line option for jumping to a specific line (e. g. BOXER) and if this option has been made known to CET during the installation of CET, the cursor jumps to the line containing the error, otherwise the cursor is in line 1 of the faulty text.

100 Environment variable CET_DRIVE not set

The environment variable CET_DRIVE must be defined in \AUTOEXEC.BAT (see \CET\INSTALL.ENG, 5. Page 1 of 8). There is no line SET CET_DRIVE=C: in \AUTOEXEC.BAT (C is the letter of the drive which CET was installed to).

101 Environment variable CET_EDITOR_JUMP must contain \$ or must be empty

The environment variable CET_EDITOR_JUMP defined in \AUTOEXEC.BAT has a wrong value.

102 Could not create

The file could not be created. Possible causes:

- There is no space left on the storage device.
- The storage device is write protected.
- The file exists and is write protected or the access to the file is blocked by another process.

103 Could not delete

The file could not be deleted. Possible cause:

- The file is write protected or the access to the file is blocked by another process.

104 Disk read error

An error occurred while reading from the storage device.

105 Disk write error

An error occurred while writing to the storage device. Possible causes:

- There is no space left on the storage device.
- The storage device is write protected.

106 Input File is empty

An empty text file cannot be processed by CET.

107 Missing beginning of font command

<[corresponding to]> is missing.

Example: Times m m n 10 pt 10 pt]>

108 File access denied

The access to the file was denied. Possible cause:

- Another process has exclusive access to the file.

109 File not accessible

A file which is part of CET could not be found.

- 110 **File not found**
The file could not be found.
- 111 **General I/O error**
An input/output error occurred. The storage device may be defective.
- 112 **A letter must follow an accent in Greek mode**
Greek spiritus and Greek accents (<, >, ' , ' , ~) must be followed by a letter.
Example: <GB> <o l' ogos <GE>
- 113 **Greek mode within Latin mode not allowed**
<GB>...<GE> is invalid within <LTB>...<LTE>.
Example: <LTB>in <GB>eo<GE> quod<LTE>
- 114 **Hardware failure**
An error occurred while reading from or writing to a storage device.
- 115 **Nesting of include files not allowed**
A file processed due to <IFB>...<IFE> must not contain <IFB>...<IFE>.
- 116 **Include file not found**
The file specified within <IFB>...<IFE> could not be found.
- 117 **Internal error**
An internal processing error occurred. Please report this error to the author of CET (see APPENDIX F). Put <C> at the beginning of the paragraph in which the error occurred and Typeset the text again. The error should not occur any more, as CET does not process this paragraph any more due to the comment command <C>.
- 118 **Invalid number of columns (found/expected)**
The number of columns defined with <SWB>...<SWE> (the number of the proportional widths of the columns) does not match the number of columns corresponding to <SYB>...<SYE>.
Example:
<SWB>30 30 30<SWE>
<SYB>column 1<SYS>column 2<SYE>
- 119 **Sum of column widths exceeds 100%**
The sum of the proportional widths of the columns defined with <SWB>...<SWE> must not be greater than 100.
Example: <SWB>40 40 40<SWE>
- 120 **Invalid column width**
A proportional column width defined with <SWB>...<SWE> must not exceed 100 per cent.
Example: <SWB>110 10<SWE>
- 121 **Invalid command in headline or sigla text**
This command is invalid within <HLB>...<HLE>, <HRB>...<HRE> and <S*B>... <S*E> (* must be replaced with 1 ... 9).
Example: <HLB><GB><o l' ogos<GE><HLE>
- 122 **Invalid display locked line number**
The only valid options are: first, last, all
Example: DISPLAY_LOCKED_LINE_NUMBER *bad
- 123 **Invalid filename extension**
The filename extension is invalid. Change the filename extension. The following extensions are invalid: .AUX .BAT .CFG .DVI .END .ERR .FIP .FIT .FNT .IM .IMF .IMS .IMT .IW .IWF .IWS .LBL .LBT .LOG .PS .REF .RET .STY .TEX
- 124 **Missing end of filenote**
The end of a filenote is missing (1>, ... , 9>).
Example: <1 filenote 1
- 125 **Missing beginning of filenote**
The beginning of a filenote is missing (<1, ... , <9).
Example: filenote 1>
- 126 **Missing end of font command**
> corresponding to <[is missing.
Example: <[Times m m n 10 pt 10 pt
- 127 **Syntax error in font file**
&CET&CEPP.FNT ist syntactically wrong. Delete the file &CET&CEPP.FNT (CET creates a new file CEPP.FNT if CEPP.FNT does not exist). Remove all lines after the first \typeout{Loading ... from the file &EMTEX&TEXINPUT&PSFONTS.TEX. Remove all lines after rprerc Courier from the file &EMTEX&PS&PSFONTS.MAP. Reinstall all PostScript fonts with addpsfnt (see Chapter 4).

- 128 **Font size unit must be pt**
The only valid unit is pt.
- 129 **Invalid headline position**
The only valid options are: inner, center, outer
Example: HEADLINE_POSITION *bad
- 130 **Invalid hyphenation rules**
The only valid options are: German, French, Latin, English, None, Spare1, Spare2
Example: HYPHENATION_RULES *Sanskrit
- 131 **Invalid include file command**
<IFB>...<IFE> must be in its own paragraph.
Example: <IFB>chapter1.txt<IFE> test
- 132 **Invalid within filenotes, alternate lemmata and variant readings**
Labels are invalid within filenotes, alternative lemmata and variant readings.
Example: (*lemma*) {variant reading @label1}
- 133 **Invalid label name**
The only valid characters within a label name are: a ... z, A ... Z, 0 ... 9. Spaces within a label name are invalid.
Example: @label#1
- 134 **Invalid lineation type**
The only valid options are: page, section
Example: LINEATION_BY *bad
- 135 **Invalid lineation margin**
The only valid options are: left, right, inner, outer
Example: LINEATION_MARGIN *bad
- 136 **Invalid note format**
The only valid options are: normal, paragraph, twocol, threecol
Example: FOOTNOTE1_FORMAT *bad
- 137 **Invalid note number**
The number of the footnote series must be within the range 1 ... 9.
Example: FOOTNOTE0_BEGIN (*
- 138 **Invalid no/yes token**
The only valid options are: Y, N
Example: FOOTNOTE1_LINE_NUMBER_OMIT *X
- 139 **Invalid number**
The number is invalid.
Example: <TWB>51<TWE>
- 140 **Invalid PostScript font**
The name of the PostScript font is unknown to CET. Non-standard PostScript fonts must be installed with addpsfnt (see Chapter 4). The available PostScript fonts can be found in the file \CET\CEPP.FNT.
Example: <[Time m m n 10 pt 10 pt]>
- 141 **Invalid shape**
The only valid options are: n, it, sl, sc, u
Example: <[Times m m bad 10 pt 10 pt]>
- 142 **Letter for marking overlapping lemmata must be unique within a paragraph**
Characters after and in front of ^ must be unique within a paragraph.
Example: (*^# (*^# #^*) { } #^*) { }
- 143 **Note series of overlapping lemma does not match**
Same characters in front of and after ^ must belong to the same note series.
Example: (*^# #^+) { }
- 144 **Invalid synopsis**
The commands for typesetting text side by side contain an error.
Examples: <SYB> within <SYB>...<SYE>; <SYB> without <SYE>; <SYE> without <SYB>; <SYS> without <SYB>; ...
- 145 **Invalid text after alternate lemma**
Only spaces are valid between the end of an alternative lemma and the beginning of the variant reading.
- 146 **Invalid text after lemma**
The only valid terms after a lemma are: spaces, alternative lemma, variant reading

- 147 **Invalid text position**
 The only valid options are: left, right
 Example: FILENOTE1_TEXT_POSITION *bad
- 148 **\ not allowed. Embed TeX commands within <TMB> ... <TME>**
 TeX commands start with \. If you know what you are doing, you can use TeX commands within <TMB>...<TME>.
- 149 **Transparency not allowed after GLOBAL_FONT**
 At the beginning of the text GLOBAL_FONT is active. As this is the first active font, the font must be completely defined.
 Example: GLOBAL_FONT <[Times - - - 12 pt 12 pt]>
- 150 **Missing beginning of two columns mode**
 <TCB> corresponding to <TCE> is missing.
- 151 **Invalid unit**
 The only valid units are: pt, pc, in, bp, cm, mm, dd, cc, em, ex (in some cases em and ex are invalid)
 Example: <[Times - - - 12 xx 12 xx]>
- 152 **Invalid value**
 The value must not be negative. In some cases the value must not equal 0.
 Example: <[Times m m n 1x pt 12 pt]>
- 153 **Invalid weight class**
 The only valid weight classes are: ul, el, l, sl, m, sb, b, eb, ub
 Example: <[Times bad m n 10 pt 12 pt]>
- 154 **Invalid combination of weight and width**
 If the font width is transparent the font weight must be transparent as well (and vice versa).
 Example: <[Times m - n 12 pt 12 pt]>
- 155 **Invalid width class**
 The only valid width classes are: uc, ec, c, sc, m, sx, x, ex, ux
 Example: <[Times m bad n 10 pt 12 pt]>
- 156 **Latin mode already active**
 <LTB> within <LTB>...<LTE> or outside Greek text is invalid.
 Example: <LTB>in <LTB>eo<LTE> quod<LTE>
- 157 **Lineation is off; notes and labels not allowed.**
 Footnotes, endnotes and labels are invalid outside unnumbered text. The line numbering must be switch on with #L+ or #N+ #L+ (note that #N+ and #L+ are effective beginning with the **following** paragraph).
- 158 **Missing end of alternate lemma**
 -} corresponding to {- is missing.
 Example: (*lemma*) {-alternative lemma {variant reading}}
- 159 **At least two column widths must be specified**
 <SWB>...<SWE> must contain at least two column widths.
 Example: <SWB>40<SWE>
- 160 **Matching command does not follow**
 The corresponding command could not be found.
 Example: <IB>text
- 161 **Expected but not found**
 The text indicated in the error message could not be found.
 Example: @[label1]
- 162 **Missing variant reading**
 Footnotes and endnotes must contain a variant reading.
 Examples:
 (*lemma*) text
 (*lemma*) x {var.}
 (*lemma*) {-alt. lemma-} x {var.}
- 163 **Missing end of variant reading**
 } corresponding to { is missing.
 Example: (*lemma*) {var.}
- 164 **Two (or more) options marked**
 Only one option may be marked with *.
 Example: LINEATION_BY *page *section

- 165 **No font available**
The file \CET\CEPP.FNT does not contain a font. For a solution to this problem see error message "127 Syntax error in font file".
- 166 **No option marked**
Exactly one option must be marked with *.
Example: `LINEATION_BY page section`
- 167 **Out of memory**
The input file is too complex. Try to increase the main memory by removing unnecessary drivers and programs from \AUTOEXEC.BAT and \CONFIG.SYS.
- 168 **Paragraph too long**
The paragraph is too long (i. e., contains more than about 64000 characters). Break the paragraph into several smaller ones by inserting empty lines.
- 169 **Duplicate token**
Command names must be unique.
Example:
`FOOTNOTE1_BEGIN <FNB>`
`FOOTNOTE2_BEGIN <FNB>`
- 170 **Matching note token not found**
The character corresponding to the character after/in front of ^ is missing.
Example: `(*^# { }`
- 171 **Missing end of synopsis width**
<SWE> corresponding to <SWB> is missing.
Example: `<SWB>10 10 10`
- 172 **Synopsis not initialised.**
Before you can use <SYB>...<SYE> you have to initialize the column widths with <SWB>...<SWE>.
- 173 **Syntax: cepp <filename>**
CEPPEXE was called with wrong command line parameters.
- 174 **Syntax error**
The current line is syntactically wrong.
Example: `XXX` (as a line in the configuration file)
- 175 **Command name too long**
The length of a command name must not exceed 10 characters.
Example: `LATIN_BEGIN 12345678901`
- 176 **Too many columns**
Up to 9 columns may be defined with <SWB>...<SWE>.
Example: `<SWB>1 2 3 4 5 6 7 8 9 10<SWE>`
- 177 **Two columns mode not initialised**
Before you can use <TCB>...<TCE>, you have to define the column width with <TWB>...<TWE>.
- 178 **Missing end of note**
) , + , ... * , ... corresponding to (* , (+ , ... [* , ... is missing.
Example: `(*lemma {variant reading}`
- 179 **Missing beginning of lemma**
(* , (+ , ... [* , ... corresponding to * , + , ... * , ... is missing.
Example: `lemma*) {variant reading}`
- 180 **Missing matching marker for overlapping lemma**
The character corresponding to the character after/in front of ^ is missing.
Example: `(*^# =^*) { }`
- 181 **Missing matching command**
The corresponding command is missing.
Example: `bold<BDE>`
- 182 **Word too long**
The word is too long. Insert spaces into the word.
- 183 **Two columns mode already active**
The two columns mode is already active due to a previous <TCB>.

- 184 **Missing end of two columns mode**
<TCE> corresponding to <TCB> is missing.
- 185 **<OM> and <!> invalid outside variant readings**
<OM> and <!> are invalid outside variant readings.
Example: (*<!>lemma*) {var. }
- 186 **Command overlaps**
Corresponding command names must not overlap other corresponding command names.
Example: <IB> . . . <GB> . . . <IE> . . . <GE>
- 187 **Matching command is on a different level**
Corresponding command names must not overlap lemmata, alternative lemmata and variant readings.
Example: (*lemma <IB>A*) {var. } <IE>
- 188 **Line numbering is already active**
Line numbering is already active due to a previous #N+.
Example: #N+ #N+
- 189 **Line numbering is off**
Line numbering has not been activated with #N+.
Example: #L+ without previous #N+

F. The authors of the CET components

CET consists of several software packages. There were no CET without the work of the various authors listed below.

The components of CET, the names and the addresses of the authors and references to correspondig documentation are listed in the following table:

Program	Author	Address	Documentation
CET shell, CET preprocessor	Bernt Karasch	Heinrich-König-Str. 18, 44797 Bochum, Germany bernt.karasch@rz.ruhr-uni-bochum.de	<code>\CET\REFERENZ.DOK</code>
EDMAC	John Lavagnino	Department of English and American Literature, Brandeis University, 415 South Street, Waltham, MA 02254-9110, USA lav@binah.cc.brandeis.edu	<code>\EMTEX\TEXINPUT\EDMAC.DOC</code>
	Dominik Wujastyk	Wellcome Institute for the History of Medicine, 183 Euston Road, London NW1 2BE, UK d.wujastyk@ucl.ac.uk	
LaTeX2e	LaTeX3 project		
em _T E _X	Donald E. Knuth Eberhard Mattes	Stanford University Teckstraße 81, 71696 Möglingen, Germany mattes@azu.informatik.uni-stuttgart.de	<code>\EMTEX\DOC*.*</code>
Hyphenation rules (German)	Norbert Schwarz	Rechenzentrum Ruhr-Universität Bochum, Universitätsstr. 150, 44780 Bochum, Germany norbert.schwarz@rz.ruhr-uni-bochum.de	
	Bernd Raichle	Stettener Str. 73, 73732 Esslingen, Germany raichle@informatik.uni-stuttgart.de	
Hyphenation rules (Latin)	Claudio Beccari	Politecnico di Torino, Torino, Italy beccari@polito.it	
Hyphenation rules (English)	Dominik Wujastyk Graham Toal	Wellcome Institute for the History of Medicine, 183 Euston Road, London NW1 2BE, UK d.wujastyk@ucl.ac.uk	
RSX	Rainer Schnither	rainer@mathematik.uni-bielefeld.de	<code>\CET\COPYING.RSX</code>
emx	Eberhard Mattes	Teckstraße 81, 71696 Möglingen, Germany mattes@azu.informatik.uni-stuttgart.de	<code>\CET\COPYING.EMX</code>
emxfemu	W. Metzenthén	22 Parker St, Ormond, Vic 3163, Australia billm@vaxc.cc.monash.edu.au	<code>\CET\COPYING.EMX</code>
Levy fonts (Greek)	Silvio Levy		<code>\EMTEX\MFINPUT\GREEK*.TEX</code>

PostScript pixel fonts	Ganesh Thiagarajan Anthony Venson	gany@rtpc01.eng.lsu.edu andy@rtpc01.eng.lsu.edu	\TE\FONTS\README.PS
dvips	Tomas Rokicki Donald E. Knuth	rokicki@cs.stanford.edu Stanford University	\EMTEX\DOC\DVIPS.DOC, \EMTEX\TEXINPUT\DVIPS.TEX
Ghostscript	L. Peter Deutsch		\GS2.52*.DOC
ps2pk	Piet Tutelaers	rcpt@urc.tue.nl	\EMTEX\DOC\PS2PK*.MAN
BOXER	David R. Hamel	Boxer Software, P.O. Box 3230, Peterborough, NH 03458-3230, USA 70242.2126@compuserve.com German source: Nane Jürgensen Nordergraben 26 24937 Flensburg Phone (0461) 182340 Fax (0461) 182341 100021.414@compuserve.com	\BOXER\BOXER.DOC
chktxt	Indridi Bjornsson	indridi@rhi.hi.is	\CET\CHKTXT.DOC
Suggestions, bug reports	Olli Hallamaa J. Heinrich Riggert	Institute for Systematic Theology, P.O. Box 33, 00014 University of Helsinki, Finland ohallamaa@teologi1.helsinki.fi Universität zu Köln, Thomas-Institut, Universitätsstr. 22, 50923 Köln, Germany	

Ghostscript, RSX, emx and emxfpemu are subject to the "GNU General Public License Version 2, June 1991" (\GS2.52\COPYING). Due to this licence (see section 3.b) I have to offer you the source code for Ghostscript, RSX, emx and emxfpemu for a nominal price for three years starting with the date on your CET invoice.

G. Trademarks

The following terms are (registered) trademarks:

Name	Holder
Adobe	Adobe Systems Incorporated
Canon	Canon Kabushiki Kaisha
DeskJet	Hewlett-Packard Company
Epson	Epson Kabushiki Kaisha
Helvetica	Linotype Company
IBM	IBM Corporation
LaserJet	Hewlett-Packard Company
MS	Microsoft Corporation
MS-DOS	Microsoft Corporation
OS/2	IBM Corporation
PaintJet	Hewlett-Packard Company
PostScript	Adobe Systems Incorporated
Proprinter	IBM Corporation
Times	Linotype Company
Windows	Microsoft Corporation

All other brand and product names referred to in this document are trademarks or registered trademarks of their respective holders.

H. Commands names and keywords

1. Alphabetical list of default command names

Command	Keyword	Page			
			<1	FILENOTE1_BEGIN	39
			<2	FILENOTE2_BEGIN	39
			<3	FILENOTE3_BEGIN	39
#)	FOOTNOTES5_END	33	<4	FILENOTE4_BEGIN	39
#E	ENDNOTES_PRINT	37	<5	FILENOTE5_BEGIN	39
#E1	ENDNOTE1_PRINT	37	<6	FILENOTE6_BEGIN	39
#E2	ENDNOTE2_PRINT	37	<7	FILENOTE7_BEGIN	39
#E3	ENDNOTE3_PRINT	37	<8	FILENOTE8_BEGIN	39
#E4	ENDNOTE4_PRINT	37	<9	FILENOTE9_BEGIN	39
#E5	ENDNOTE5_PRINT	37	<=>	MACRON	27
#E6	ENDNOTE6_PRINT	37	<AE>	CAPITAL_AE	27
#E7	ENDNOTE7_PRINT	37	<ae>	SMALL_AE	27
#E8	ENDNOTE8_PRINT	37	<Ao>	CAPITAL_A_CIRCLE	27
#E9	ENDNOTE9_PRINT	37	<ao>	SMALL_A_CIRCLE	27
#K+	LINEATION_LOCK	16		BAR_UNDER	27
#K-	LINEATION_UNLOCK	16	<BDB>	BOLD_BEGIN	20
#L+	LINEATION_BEGIN	16	<BDE>	BOLD_END	20
#L-	LINEATION_END	16	<c>	CEDILLA	27
#N+	NUMBERING_BEGIN	16	<C>	COMMENT	26
#N-	NUMBERING_END	16	<CB>	CENTERED_BEGIN	20
#S+	SUB_LINEATION_BEGIN	16	<CE>	CENTERED_END	20
#S-	SUB_LINEATION_END	16	<GR>	HYPHENATION_GERMAN	15
#	ENDNOTE5_END	37	<d>	DOT_UNDER	27
\$)	FOOTNOTE6_END	33	<FR>	HYPHENATION_FRENCH	15
\$	ENDNOTE6_END	37	<GB>	GREEK_BEGIN	20
&)	FOOTNOTE7_END	33	<GE>	GREEK_END	20
&	ENDNOTE7_END	37	<H>	HUNGARUMLAUT	27
(#	FOOTNOTES5_BEGIN	33	<HAB>	HANGAFTER_BEGIN	24
(\$	FOOTNOTE6_BEGIN	33	<HAE>	HANGAFTER_END	24
(&	FOOTNOTE7_BEGIN	33	<HIB>	HANGINDENT_BEGIN	24
(*	FOOTNOTE1_BEGIN	33	<HIE>	HANGINDENT_END	24
(+	FOOTNOTE2_BEGIN	33	<HLB>	HEADLINE_TEXTL_BEGIN	14
(-	FOOTNOTE3_BEGIN	33	<HLE>	HEADLINE_TEXTL_END	14
(=	FOOTNOTE4_BEGIN	33	<HRB>	HEADLINE_TEXTR_BEGIN	14
(^	FOOTNOTE8_BEGIN	33	<HRE>	HEADLINE_TEXTR_END	14
{}	FOOTNOTE9_BEGIN	33	<HSB>	HSKIP_BEGIN	24
)	REF_SIMPLE_END	42	<HSE>	HSKIP_END	24
*)	FOOTNOTE1_END	33	<i>	DOTLESS_I	27
*	ENDNOTE1_END	37	<IB>	ITALIC_BEGIN	20
+)	FOOTNOTE2_END	33	<IE>	ITALIC_END	20
+	ENDNOTE2_END	37	<IFB>	INCLUDE_FILE_BEGIN	25
,	REF_DOUBLE_SEPARATOR	42	<IFE>	INCLUDE_FILE_END	25
-)	FOOTNOTE3_END	33	<L/>	CAPITAL_L_SLASH	27
-	ENDNOTE*_A_L_END	37	<l/>	SMALL_L_SLASH	27
-}	FOOTNOTE*_A_L_END	33	<LAB>	LEFT_BEGIN	20
1>	FILENOTE1_END	39	<LAE>	LEFT_END	20
2>	FILENOTE2_END	39	<LAT>	HYPHENATION_LATIN	15
3>	FILENOTE3_END	39	<LB>	LOWERED_BEGIN	20
4>	FILENOTE4_END	39	<LE>	LOWERED_END	20
5>	FILENOTE5_END	39	<LMB>	LEFT_MARGIN_BEGIN	23
6>	FILENOTE6_END	39	<LME>	LEFT_MARGIN_END	23
7>	FILENOTE7_END	39	<LTB>	LATIN_BEGIN	20
8>	FILENOTE8_END	39	<LTE>	LATIN_END	20
9>	FILENOTE9_END	39	<NB>	NORMAL_BEGIN	20
< >	OMIT_SEPARATOR	26	<NE>	NORMAL_END	20
<'>	DIERESIS	27	<NON>	HYPHENATION_NONE	15
<'>	ACUTE	27	<NP>	NEW_PAGE	24
<>	DOT	27	<O/>	CAPITAL_O_SLASH	27

<o/>	SMALL_O_SLASH	27	<TWE>	TWO_COL_WIDTH_END	22
<OE>	CAPITAL_OE	27	<u>	BREVE	27
<oe>	SMALL_OE	27	<UB>	UNDERLINED_BEGIN	20
<OM>	OMIT	26	<UE>	UNDERLINED_END	20
<PIB>	INDENT_BEGIN	24	<UIB>	UPRIGHT_ITALIC_BEGIN	20
<PIE>	INDENT_END	24	<UIE>	UPRIGHT_ITALIC_END	20
<PNB>	PAGE_NUMBER_BEGIN	14	<UK>	HYPHENATION_ENGLISH	15
<PNE>	PAGE_NUMBER_END	14	<v>	CHECK	27
<PSB>	PARAGRAPH_SKIP_BEGIN	24	<VSB>	VSKIP_BEGIN	24
<PSE>	PARAGRAPH_SKIP_END	24	<VSE>	VSKIP_END	24
<RAB>	RIGHT_BEGIN	20	<[FONT_BEGIN	23
<RAE>	RIGHT_END	20	<>	CIRCUMFLEX	27
<RB>	RAISED_BEGIN	20	<^>	GRAVE	27
<RE>	RAISED_END	20	<~>	TILDE	27
<RMB>	RIGHT_MARGIN_BEGIN	23	=)	FOOTNOTE4_END	33
<RME>	RIGHT_MARGIN_END	23	=]	ENDNOTE4_END	37
<S/>	SECTION	27	@	REF_LABEL_BEGIN	42
<\$1B>	FOOTNOTE1_SIGL_BEGIN	34	@(REF_SIMPLE_BEGIN	42
<\$1E>	FOOTNOTE1_SIGL_END	34	@[REF_DOUBLE_BEGIN	42
<\$2B>	FOOTNOTE2_SIGL_BEGIN	34	#[ENDNOTE5_BEGIN	37
<\$2E>	FOOTNOTE2_SIGL_END	34	[\$	ENDNOTE6_BEGIN	37
<\$3B>	FOOTNOTE3_SIGL_BEGIN	34	[&	ENDNOTE7_BEGIN	37
<\$3E>	FOOTNOTE3_SIGL_END	34	[*	ENDNOTE1_BEGIN	37
<\$4B>	FOOTNOTE4_SIGL_BEGIN	34	[+	ENDNOTE2_BEGIN	37
<\$4E>	FOOTNOTE4_SIGL_END	34	[=	ENDNOTE4_BEGIN	37
<\$5B>	FOOTNOTE5_SIGL_BEGIN	34	[^	ENDNOTE8_BEGIN	37
<\$5E>	FOOTNOTE5_SIGL_END	34	[ENDNOTE3_BEGIN	37
<\$6B>	FOOTNOTE6_SIGL_BEGIN	34	[}	ENDNOTE9_BEGIN	37
<\$6E>	FOOTNOTE6_SIGL_END	34]	REF_DOUBLE_END	42
<\$7B>	FOOTNOTE7_SIGL_BEGIN	34]>	FONT_END	23
<\$7E>	FOOTNOTE7_SIGL_END	34	^)	FOOTNOTE8_END	33
<\$8B>	FOOTNOTE8_SIGL_BEGIN	34	^]	ENDNOTE8_END	37
<\$8E>	FOOTNOTE8_SIGL_END	34	{	ENDNOTE*_VAR_BEGIN	37
<\$9B>	FOOTNOTE9_SIGL_BEGIN	34	{	FOOTNOTE*_VAR_BEGIN	33
<\$9E>	FOOTNOTE9_SIGL_END	34	{}	FOOTNOTE9_END	33
<SB>	SLANTED_BEGIN	20	{-	ENDNOTE*_A_L_BEGIN	37
<SCB>	SMALL_CAPS_BEGIN	20	{-	FOOTNOTE*_A_L_BEGIN	33
<SCE>	SMALL_CAPS_END	20	{}	ENDNOTE9_END	33
<SE>	SLANTED_END	20	-	HYPHENATION	26
<SP1>	HYPHENATION_SPARE1	15	}	ENDNOTE3_END	37
<SP2>	HYPHENATION_SPARE2	15	}	ENDNOTE*_VAR_END	37
<SPB>	SPACED_BEGIN	20	}	FOOTNOTE*_VAR_END	33
<SPE>	SPACED_END	20			
<SVSB>	STATIC_VSKIP_BEGIN	24			
<SVSE>	STATIC_VSKIP_END	24			
<SWB>	SYNOPSIS_WIDTH_BEGIN	22			
<SWE>	SYNOPSIS_WIDTH_END	22			
<SYB>	SYNOPSIS_BEGIN	22			
<SYE>	SYNOPSIS_END	22			
<SYS>	SYNOPSIS_SEPARATOR	22			
<sz>	SZ	27			
<TB>	TRANSPARENT_BEGIN	25			
<TCB>	TWO_COL_BEGIN	22			
<TCE>	TWO_COL_END	22			
<TE>	TRANSPARENT_END	25			
<TMB>	TEX_MODE_BEGIN	25			
<TME>	TEX_MODE_END	25			
<TWB>	TWO_COL_WIDTH_BEGIN	22			

2. Alphabetical list of keywords

Name of keyword	Default	Page
ACUTE	<'>	27
APPARATUS_WORD_GLUE	0.3 em plus 0.2 em minus 0.1 em	11
BAR_UNDER		27
BOLD_BEGIN	<BDB>	20
BOLD_END	<BDE>	20
BREVE	<u>	27
CAPITAL_AE	<AE>	27
CAPITAL_A_CIRCLE	<Ao>	27
CAPITAL_L_SLASH	<L/>	27
CAPITAL_OE	<OE>	27
CAPITAL_O_SLASH	<O/>	27
CEDILLA	<c>	27
CENTERED_BEGIN	<CB>	20
CENTERED_END	<CE>	20
CHECK	<v>	27
CIRCUMFLEX	<^>	27
COMMENT	<C>	26
CROP_MARK_BACK_MARGIN	2.5 cm	13
CROP_MARK_GAP	5.0 pt	12
CROP_MARK_HEAD_MARGIN	1.9 cm	13
CROP_MARK_HORIZONTAL_DISTANCE	16.1 cm	13
CROP_MARK_VERTICAL_DISTANCE	23.4 cm	13
CROP_MARK_WIDTH	0.4 pt	12
CUT_LINE_NUMBER	*N Y	17
CUT_SUBLINE_NUMBER	*N Y	17
DIERESIS	<">	27
DISPLAY_LOCKED_LINE_NUMBER	*first last all	16
DOT	<.>	27
DOTLESS_I	<i>	27
DOT_UNDER	<d>	27
ENDNOTE*_A_L_BEGIN	{-	37
ENDNOTE*_A_L_END	-}	37
ENDNOTE*_BEGIN	s. Text	37
ENDNOTE*_END	s. Text	37
ENDNOTE*_LEMMA_ABBREVIATE	N *Y	37
ENDNOTE*_LEMMA_FONT	- - - - 10.0 pt 10.0 pt	37
ENDNOTE*_LEMMA_LOWER_CASE	*N Y	38
ENDNOTE*_LEMMA_OMIT	*N Y	37
ENDNOTE*_LINE_NUMBER_FONT	Times m m n 10.0 pt 10.0 pt	37
ENDNOTE*_LINE_NUMBER_OMIT	*N Y	37
ENDNOTE*_PRINT	s. Text	37
ENDNOTE*_SEPARATOR]	38
ENDNOTE*_SEPARATOR_FONT	Times m m n 10.0 pt 10.0 pt	38
ENDNOTE*_VARIANT_FONT	- - - - 10.0 pt 10.0 pt	38
ENDNOTE*_VAR_BEGIN	{	37
ENDNOTE*_VAR_END	}	37
ENDNOTES_PRINT	#E	37
FILENOTE*_BEGIN	s. Text	39
FILENOTE*_END	s. Text	39
FILENOTE*_FILENAME	s. Text	39
FILENOTE*_TEXT_POSITION	*left right	39
FONT_BEGIN	<[23
FONT_END	23	
FOOTNOTE*_A_L_BEGIN	{-	33
FOOTNOTE*_A_L_END	-}	33
FOOTNOTE*_BEGIN	s. Text	33
FOOTNOTE*_END	s. Text	33
FOOTNOTE*_FORMAT	normal *paragraph twocol threecol	32

FOOTNOTE*_LEMMA_ABBREVIATE	N *Y	34
FOOTNOTE*_LEMMA_FONT	- - - - 10.0 pt 10.0 pt	33
FOOTNOTE*_LEMMA_LOWER_CASE	*N Y	34
FOOTNOTE*_LEMMA_OMIT	*N Y	33
FOOTNOTE*_LEMMA_SEPARATOR		34
FOOTNOTE*_LEMMA_SEPARATOR_FONT	Times m m n 10.0 pt 10.0 pt	34
FOOTNOTE*_LINE_NUMBER_FONT	Times m m n 10.0 pt 10.0 pt	33
FOOTNOTE*_LINE_NUMBER_OMIT	*N Y	33
FOOTNOTE*_LINE_NUMBER_REPEAT	N *Y	34
FOOTNOTE*_RULE	2 in 0.4 pt	32
FOOTNOTE*_SEPARATOR]	34
FOOTNOTE*_SEPARATOR_FONT	Times m m n 10.0 pt 10.0 pt	34
FOOTNOTE*_SIGLA	*N Y	34
FOOTNOTE*_SIGLA_DISTANCE	4.0 em	34
FOOTNOTE*_SIGLA_FONT	Times m m n 10.0 pt 10.0 pt	34
FOOTNOTE*_SIGL_BEGIN	s. Text	34
FOOTNOTE*_SIGL_END	s. Text	34
FOOTNOTE*_VARIANT_FONT	- - - - 10.0 pt 10.0 pt	34
FOOTNOTE*_VAR_BEGIN	{	33
FOOTNOTE*_VAR_END	}	33
FOOTNOTE*_PARAGRAPH_GLUE	1.0 em plus 0.4 em minus 0.4 em	11
FRENCH_SPACING	N *Y	11
GLOBAL_FONT	Times m m n 12.0 pt 12.0 pt	11
GRAVE	<'>	27
GREEK_BEGIN	<GB>	20
GREEK_END	<GE>	20
HANGAFTER_BEGIN	<HAB>	24
HANGAFTER_END	<HAE>	24
HANGINDENT_BEGIN	<HIB>	24
HANGINDENT_END	<HIE>	24
HEADLINE_FONT	Times m m n 10.0 pt 10.0 pt	13
HEADLINE_POSITION	inner *center outer	13
HEADLINE_TEXTL_BEGIN	<HLB>	14
HEADLINE_TEXTL_END	<HLE>	14
HEADLINE_TEXTR_BEGIN	<HRB>	14
HEADLINE_TEXTR_END	<HRE>	14
HEADLINE_TEXT_LEFT		14
HEADLINE_TEXT_RIGHT		14
HEADLINE_VERTICAL_DISTANCE	0.4 cm	13
HORIZONTAL_OFFSET	-0.3 cm	10
HORIZONTAL_SIZE	11.0 cm	10
HORIZONTAL_TOLERANCE	0.1 pt	11
HSKIP_BEGIN	<HSB>	24
HSKIP_END	<HSE>	24
HUNGARUMLAUT	<H>	27
HYPHENATION	-	26
HYPHENATION_ENGLISH	<UK>	15
HYPHENATION_FRENCH	<FR>	15
HYPHENATION_GERMAN	<GR>	15
HYPHENATION_LATIN	<LAT>	15
HYPHENATION_NONE	<NON>	15
HYPHENATION_RULES	German French *Latin English None	14
HYPHENATION_SPARE1	<SP1>	15
HYPHENATION_SPARE2	<SP2>	15
INCLUDE_FILE_BEGIN	<IFB>	25
INCLUDE_FILE_END	<IFE>	25
INDENT_BEGIN	<PIB>	24
INDENT_END	<PIE>	24
ITALIC_BEGIN	<IB>	20
ITALIC_END	<IE>	20
LATIN_BEGIN	<LTB>	20
LATIN_END	<LTE>	20

2. Alphabetical list of keywords

73

LATIN_FONT	Times m m n 12.0 pt 12.0 pt	11
LEFT_BEGIN	<LAB>	20
LEFT_END	<LAE>	20
LEFT_MARGIN_BEGIN	<LMB>	23
LEFT_MARGIN_END	<LME>	23
LINEATION_BEGIN	#L+	16
LINEATION_BY	page *section	15
LINEATION_END	#L-	16
LINEATION_LOCK	#K+	16
LINEATION_MARGIN	left right *inner outer	15
LINEATION_UNLOCK	#K-	16
LINE_NUMBER_DISTANCE	1.0 pc	15
LINE_NUMBER_FIRST	5	15
LINE_NUMBER_FONT	Times m m n 10.0 pt 10.0 pt	15
LINE_NUMBER_INCREMENT	5	15
LOWERED_BEGIN	<LB>	20
LOWERED_END	<LE>	20
MACRON	<=>	27
MARK_BAD_LINES	N *Y	11
NEW_PAGE	<NP>	24
NORMAL_BEGIN	<NB>	20
NORMAL_END	<NE>	20
NUMBERING_BEGIN	#N+	16
NUMBERING_END	#N-	16
OMIT	<OM>	26
OMIT_SEPARATOR	<!>	26
PAGE_NUMBER_FONT	Times m m n 10.0 pt 10.0 pt	14
PAGE_NUMBER_BEGIN	<PNB>	14
PAGE_NUMBER_END	<PNE>	14
PARAGRAPH_INDENTATION	1.0 cm	11
PARAGRAPH_SKIP_BEGIN	<PSB>	24
PARAGRAPH_SKIP_END	<PSE>	24
RAISED_BEGIN	<RB>	20
RAISED_END	<RE>	20
REF_DOUBLE_BEGIN	@[42
REF_DOUBLE_END]	42
REF_DOUBLE_SEPARATOR	,	42
REF_DOUBLE_TEXT	p._	43
REF_DOUBLE_TEXT_ABBREVIATE	N *Y	43
REF_DOUBLE_TEXT_AFTER_LINE	.	43
REF_DOUBLE_TEXT_AFTER_PAGE	,	43
REF_DOUBLE_TEXT OMIT SAME PAGE	N *Y	43
REF_DOUBLE_TEXT SAME PAGE	l._	43
REF_DOUBLE_TEXT_SEPARATOR	-	43
REF_LABEL_BEGIN	@	42
REF_LABEL_END		42
REF_SIMPLE_BEGIN	@(42
REF_SIMPLE_END)	42
REF_SIMPLE_TEXT	p. %p,%l.%s	42
REF_SIMPLE_TEXT SAME PAGE	l. %l.%s	42
RIGHT_BEGIN	<RAB>	20
RIGHT_END	<RAE>	20
RIGHT_MARGIN_BEGIN	<RMB>	23
RIGHT_MARGIN_END	<RME>	23
SECTION	<S/>	27
SLANTED_BEGIN	<SB>	20
SLANTED_END	<SE>	20
SMALL_AE	<ae>	27
SMALL_A_CIRCLE	<ao>	27
SMALL_CAPS_BEGIN	<SCB>	27
SMALL_CAPS_END	<SCE>	27
SMALL_L_SLASH	<L/>	27

SMALL_OE	<oe>	27
SMALL_O_SLASH	<o/>	27
SPACED_BEGIN	<SPB>	20
SPACED_END	<SPE>	20
SPACE_OUT_DISTANCE	0.2 em	11
STATIC_VSKIP_BEGIN	<SVSB>	24
STATIC_VSKIP_END	<SVSE>	24
SUB_LINEATION_BEGIN	#S+	16
SUB_LINEATION_END	#S-	16
SUB_LINE_NUMBER_FIRST	5	15
SUB_LINE_NUMBER_INCREMENT	5	15
SYNOPSIS_BEGIN	<SYB>	22
SYNOPSIS_END	<SYE>	22
SYNOPSIS_SEPARATOR	<SYS>	22
SYNOPSIS_WIDTH_BEGIN	<SWB>	22
SYNOPSIS_WIDTH_END	<SWE>	22
SZ	<sz>	27
TEX_MODE_BEGIN	<TMB>	25
TEX_MODE_END	<TME>	25
TILDE	<~>	27
TRANSPARENT_BEGIN	<TB>	25
TRANSPARENT_END	<TE>	25
TWO_COL_BEGIN	<TCB>	22
TWO_COL_END	<TCE>	22
TWO_COL_WIDTH_BEGIN	<TWB>	22
TWO_COL_WIDTH_END	<TWE>	22
UNDERLINED_BEGIN	<UB>	20
UNDERLINED_END	<UE>	20
UPRIGHT_ITALIC_BEGIN	<UIB>	20
UPRIGHT_ITALIC_END	<UIE>	20
VERTICAL_OFFSET	1.5 cm	10
VERTICAL_SIZE	17.0 cm	10
VSKIP_BEGIN	<VSB>	24
VSKIP_END	<VSE>	24
WORD_DISTANCE	20	11
WORD_GLUE	0.3 em plus 0.2 em minus 0.1 em	11

INDEX

The default command names and the keywords are listed in Appendix H and are not contained in this index.

*.DVI	5f.	crop mark	12f.
*.IM	45f.	dash	20
*.IMF	45f.	diaeresis	21, 27
*.IMT	45f.	double acute accent	27
*.IW	44f.	DVI file	3
*.IWF	44f.	dvips	3, 67
*.PS	5f.	DVIPS32.EXE	5f.
*.STY	4f.	EDMAC	3f., 66
*.TEX	4f.	emT _E X	66
æ	27	endnote	32, 37f.
Æ	27	endnote series	37f.
Œ	27	endnote series, printing an	37
œ	27	endnote, example	37
ß	27	endnote, position of an	32
T _E X mode	25	ERROR	47
T _E X	3f.	error messages (addpsfnt)	31
abbreviating lemmata	34, 37	error messages (Typeset)	60ff.
accent	27	exercise 1	6
acute accent	21, 27	exercise 2	12
adding a font	30f.	exercise 3	13
addpsfnt	30f.	exercise 4	14
Adobe Type 1 font	30f.	exercise 5	15
AFM file	30f.	exercise 6	17
alternative lemma	33, 37	exercise 7	20
apparatus, critical	32ff.	exercise 8	23
author index	38	exercise 9	24
authors	66f.	exercise 10	36
bar accent at the bottom	27	exercise 11	45
bar accent on the top	27	exercise 12	46
black rectangle	11, 59	exponent	11f., 20
bold	20	file name	53
BOXER	3f., 67	filenote	32, 38f.
cedilla	27	filenote series	38f.
centered	20	font	50
CEPP.EXE	3f.	font family	30
circle, A with	27	font selection	23
circle, a with	27	font, adding a	30f.
circumflex	21, 27	font, Adobe Type 1	30f.
command name	53, 69f.	font, global	11
comment	26	font, Latin	11
configuration file	4, 6, 10	font, selecting a	23
critical apparatus	32ff.	footnote	32ff.

footnote series	32ff.	lemma	33, 37
footnote series, formatting a	32	lemma separator	34
footnote, across several paragraphs	35, 38	lemma, abbreviating a	34, 37
footnote, example	33	lemma, alternative	33, 37
footnote, ordinary	34f.	lemma, suppressing a	34, 37
footnote, overlapping	35, 38	Levy font	21, 66
footnote, overlapping but unnested	35, 38	LevyGreek	21, 66
Ghostscript	3, 5, 67	limitations	49
grave accent	21, 27	line number, repeating a	34
Greek text	20f.	line number, suppressing a	34, 37
GS386.EXE	5f.	line numbering	15ff.
háček	27	macron	27
headline	13f.	margin	23f.
header, running	13f.	normal text	20
height of the text	10	number	53
help in case of problems	59	numbering, line	15ff.
Hungarian umlaut	27	o with /	27
hyphen	20	O with /	27
hyphen between numbers	20	ordinary footnote	34f.
hyphenation	26	page break	24
hyphenation rules	14f., 66	page formatting	10ff.
i without dot	27	page number	14
image area	10ff.	paragraph	20
including text files	25	paragraph indentation	11, 24
indentation	23f.	parameter type	50
index (text formatting)	20	PFB file	30f.
index of authors	38	point accent at the bottom	27
index of labels	47	point accent on the top	27
index of references	47	pointless i	27
index of words from main text	44f.	PostScript font	30f.
index of words from variant readings	45f.	PostScript font family	30
introduction	1ff.	printing an endnote series	37
iota subscriptum	21	printing endnotes	37
italic text	20	problems	59
keyword	71ff.	rectangle, black	11, 59
l with -	27	reference index	47
L with -	27	reference, from-to	42f.
label	42f.	reference, simple	42f.
label index	47	right-aligned text	20
label name	42f., 53	running header	13f.
LaTeX2e	3f., 66	section mark	27
left-aligned text	20	selection	52

INDEX

77

semicircle accent	27	tilde	27
separator line between footnote series	32	trademark	68
separator, suppressing the	26	transparent text	25
shifting the text on the page	10f.	two columns (text formatting)	22
side by side (text formatting)	21f.	umlaut	27
siglum	34	umlaut, Hungarian	27
slanted text	20	underlined text	20
small capital letters	20	unit	50
solutions to the exercises	54ff.	upright-italic	20
space	24, 53	variant reading	33, 37
space at the end of a sentence	11	width of the text	10
space between footnotes	11	word processing program	4
space between words	11	yes/no	52
spaced out	11, 20		
special character	27		
spiritus asper	21		
spiritus lenis	21		
standard PostScript fonts	50		
SUMMA.TXT	8, 18, 28, 40		
suppressing lemmata	34, 37		
suppressing line numbers	34, 37		
suppressing the separator	26		
symbolic reference	42f.		
TEX386.EXE	5		
text	52		
text formatting	20ff.		
text, bold	20		
text, centered	20		
text, Greek	20f.		
text, index of words from	44f.		
text, italic	20		
text, left-aligned	20		
text, normal	20		
text, right-aligned	20		
text, side by side	21f.		
text, slanted	20		
text, spaced out	20		
text, synoptic	21f.		
text, transparent	25		
text, two columns	22		
text, underlined	20		
text, upright-italic	20		