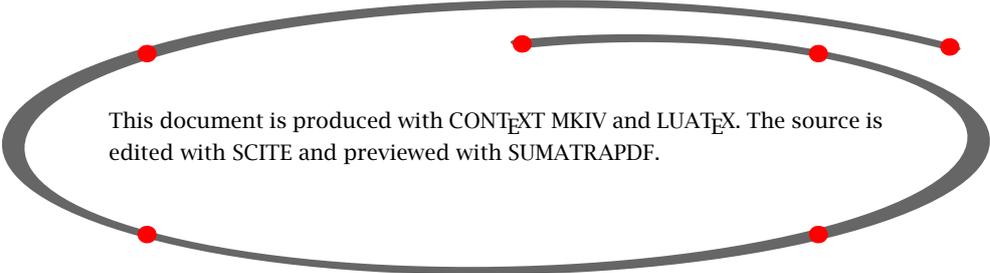


Context Mark IV

an excursion

English
version

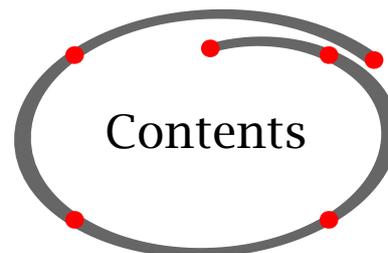
Ton Otten
PRAGMA ADE



This document is produced with $\text{CON}\text{T}_\text{E}\text{X}\text{T}$ MKIV and $\text{LUA}\text{T}_\text{E}\text{X}$. The source is edited with SCITE and previewed with SUMATRAPDF.



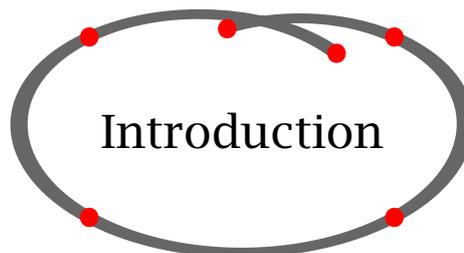
© 1991–2015 PRAGMA ADE, Ridderstraat 27, 8061GH Hasselt, The Netherlands, www.pragma-ade.com



Contents	0	34 Composite characters	83
Introduction	3	35 Page layout	83
1 How to create a textbook	5	36 Backgrounds in page areas	89
2 How to process a file	6	37 Background in paragraphs	90
3 Special characters	7	38 Paragraph spacing	91
4 Defining a document	8	39 Defining commands / macros	95
5 Setup commands	10	40 Miscellaneous	97
6 Heads	11	41 Using modules	114
7 Itemize	14	42 Presentations	115
8 Typesetting math	18	43 Graphical extension / METAPOST	116
9 Chemical stuff	26	44 User specifications	117
10 Units	27	A Command definitions	119
11 Bibliography	29	B Command index	135
12 Figures	29	C Subject index	139
13 Tables	34	D Support and further reading	143
14 Tabulation / Paragraph formatting	41	E Commands in math mode	145
15 Columns	43	F Problems during processing	149
16 Footnotes	46	G The SCITE text editor	151
17 Citations and quotations	48	H The context command	153
18 Definitions	49	I Auxilliary files	155
19 Numbered definitions	51		
20 Outlined text	52		
21 Outlined paragraphs	54		
22 Margin texts	56		
23 Page breaking and page numbering	57		
24 Page headers and footers	60		
25 Table of contents (lists)	61		
26 Registers	64		
27 Synonyms	65		
28 Sorted lists	66		
29 Referring to text elements	68		
30 Color	71		
31 Alignment	72		
32 Interactive mode in electronic documents	73		
33 Fonts and font switches	78		

Contents





CON_TE_XT is a document engineering system based on T_EX, a typesetting system and programming language to typeset and produce documents. This system is easy to use and enables you to make complex paper and electronic documents.

This manual describes the capabilities of CON_TE_XT MKIV, the available commands and their functionality.¹

This system is developed for practical applications: the typesetting and production of documents ranging from simple straight forward books up to very complex and advanced technical manuals and textbooks in a paper or an electronic version. This introductory manual describes the functionality necessary to apply standard text elements in a manual or textbook. CON_TE_XT, however, is capable of much more and for users who want more there are other manuals and sources available.

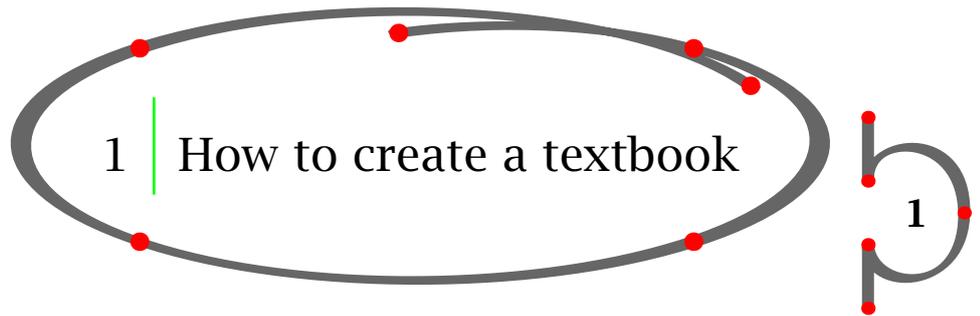
CON_TE_XT has a multi lingual interface to enable users to work with the system in their own language. This manual is available in Dutch and English.

If you want to install CON_TE_XT on your computer you can follow the installation description on the CON_TE_XT WIKI.

¹ All paper and electronic products around CON_TE_XT are produced with CON_TE_XT. All sources of these products are or will be made available electronically to give you insight in the way these products are made up.

Introduction





Let's assume you want to create a simple textbook. It has some structure and contains a title page, a few chapters, sections and sub sections. Of course there is a table of contents.

CONTEX_T can create such a document automatically if you offer the right input by means of a file. So first you have to create an input file. An input file consists of a name and an extension. You can choose any name you want but the extension has to be `.tex`. If you create a file with the name `mybook.tex` you will find no difficulties in running CONTEX_T.

An input file could look like this:

```
\starttext
\startstandardmakeup
  \midaligned{From Hasselt to America}
  \midaligned{by}
  \midaligned{J. Jonker and C. van Marle}
\stopstandardmakeup
\placecombinedlist[content]
\chapter{Introduction}
... ties between Hasselt and America ...
\chapter[rensselaer]{The Rensselaer family}
\section{The first born}
... was born in the year ...
\section{The early years}
... in those days Hasselt was ...
\section{Living and workin in America}
... life in America was ...
\chapter[lansing]{The Lansing family}
... the Lansing family was also ...
\chapter[cuyler]{The Cuyler family}
... much later Tydeman Cuyler ...
\stoptext
```

CONTEX_T expects a plain ASCII input file. Of course you can use any text-editor, as long as you save the file as standard ASCII (also called txt file) with the extension `.tex`. Note that spaces in the filename are not allowed.

The input file contains the text you want to typeset and the `CONTEXT` commands. A `CONTEXT` command begins with a backslash `\`. With the command `\starttext` you indicate the beginning of your text.

A command is sometimes followed by an argument which is enclosed by curly braces `{}`. The command `\chapter[cuyler]{The Cuyler family}` that you see in the example will have its effect on *The Cuyler family*. Its actions will have effect on the design, typography and structure. The actions may be:

1. start a new page
2. increase chapter number by one
3. place chapter number in front of chapter title
4. reserve some vertical space
5. use a big font
6. put chapter title (and page number) in the table of contents

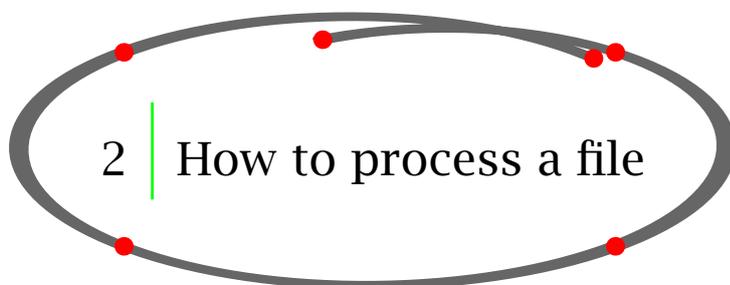
Other actions concerning running heads, number resetting and interactivity are disregarded at this moment.

Sometimes you will see two brackets `[]` directly after the command. These brackets are used to feed specific options to the command. Further on in this manual you will get more information on these brackets.

The commands in your input file can have the following appearance:

Appearance of command	Example
<code>\startcommand ... \stopcommand</code>	<code>\starttext ... \stoptext</code>
<code>\startcommand[] ... \stopcommand</code>	<code>\startitemize[packed] ... \stopitemize</code>
<code>\command</code>	<code>\item</code>
<code>\command[]</code>	<code>\in[cuyler]</code>
<code>\command{}[]</code>	<code>\at{page}[cuyler]</code>
<code>\command{}</code>	<code>\index{America}</code>
<code>\command[]{}</code>	<code>\chapter[cuyler]{The Cuyler family}</code>

If you have `CONTEXT` process the above example file, you would obtain a very simple document with a title page, a few numbered chapters and section headers and a table of content (because of `\placecombinedlist[content]`).



In this chapter we assume that you have installed and initiated `CONTEXT MKIV` correctly so that

Special characters

you can run it from the commandline in your working directory. You can find the `CONTEXT` installation procedure on the `CONTEXT` WIKI.

If you want to process a `CONTEXT` input file, you should type at the command line prompt:

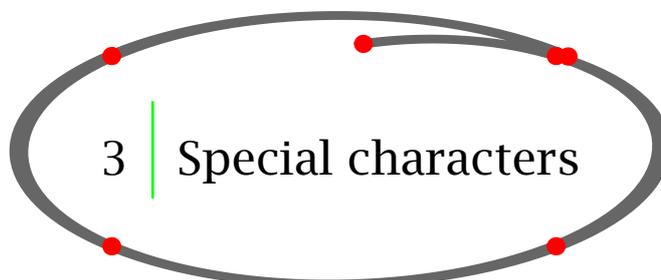
```
context myfile.tex
```

the extension `.tex` is not needed. See appendices H and I for more information on the `context` command.

After pressing `ENTER` processing will be started. `CONTEXT` will show processing information on your screen. During the processing of your input file `CONTEXT` will also inform you of what it is doing with your document. For example it will show page numbers and information about processing steps. Further more it gives warnings. These are of a typographical order and tells you when line breaking is not successful. All information on processing is stored in a `log` file that can be used for reviewing warnings and errors and the respective line numbers where they occur in your file.

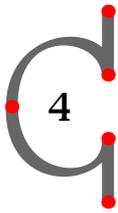
If processing is succesful the command line prompt will return and `CONTEXT` will produce the file `myfile.pdf`. The abbreviation PDF stands for Portable Document Format. This is a platform independent format for printing and viewing with `ACROBAT READER`.

When you use a configurable text editor you can also run `CONTEXT` from that editor. More information on that topic can be found appendix G.



You have seen that `CONTEXT` commands are preceded by a `\` (backslash). This means that `\` has a special meaning to `CONTEXT`. Aside from `\` there are other characters that need special attention when you want them to appear in verbatim mode or in text mode. Table 3.1 gives an overview of these special characters and what you have to type to produce them.

Other special characters have a meaning in typesetting mathematical expressions and some can be used in math mode only (see chapter 8).

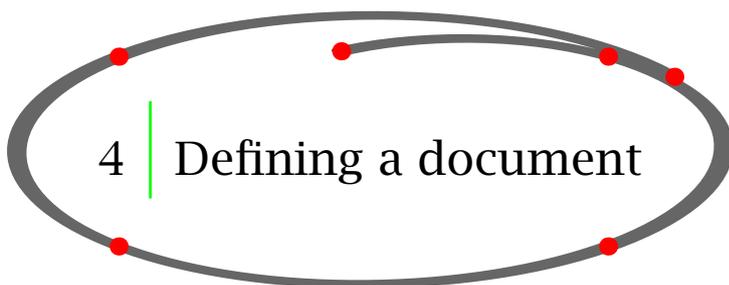


Special character		Verbatim		Text	
Character	Name	Type	Generates	Type	Generates
#	hashtag	<code>\type{#}</code>	#	<code>\#</code>	#
\$	dollar	<code>\type{\$}</code>	\$	<code>\\$</code>	\$
&	ampersand	<code>\type{&}</code>	&	<code>\&</code>	&
%	percent	<code>\type{%}</code>	%	<code>\%</code>	%
\	backslash	<code>\type{\}</code>	\	<code>\backslash</code>	\
{	right curly brace	<code>\type{+{}</code>	{	<code>\{</code>	{
}	left curly brace	<code>\type{+}</code>	}	<code>\}</code>	}
	vertical bar	<code>\type{ }</code>		<code>\ </code>	
_	underscore	<code>\type{_}</code>	_	<code>_</code>	_
~	tilde	<code>\type{~}</code>	~	<code>\lettertilde</code>	~
^	caret	<code>\type{^}</code>	^	<code>\letterhat</code>	^

Table 3.1 Special characters (1).

Special character	Verbatim		Text	
	Type	Generates	Type	Generates
+	<code>\type{+}</code>	+	<code>+\$ \$</code>	+
-	<code>\type{-}</code>	-	<code>-\$ \$</code>	-
=	<code>\type{=}</code>	=	<code>=\$ \$</code>	=
<	<code>\type{<}</code>	<	<code>\$<\$</code>	<
>	<code>\type{>}</code>	>	<code>\$>\$</code>	>

Table 3.2 Special characters (2).



Every document is started with `\starttext` and closed with `\stoptext`. All textual input is placed between these two commands and `CONTEXT` will only process that information.

Setup information is placed in the set up area just before `\starttext`.

```

\setupbodyfont[12pt]           setuparea of document
\starttext
This is a one line document.   your text
    
```



Defining a document

```
\stoptext
```

The definition of a (very simple) book could look something like this:

```
\starttext
\startstandardmakeup
  \midaligned{From Hasselt to America}
  \midaligned{by}
  \midaligned{J. Jonker and C. van Marle}
\stopstandardmakeup
\title{Foreword}
\chapter{Introduction}
\chapter{The Rensselaer family}
\chapter{The Lansing family}
\chapter{The Cuyler family}
\chapter{Appendix: Photos}
\stoptext
```

CONTEX_T comes with a predefined overall structure in which the document is divided into four main document divisions:²

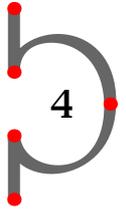
1. front matter
2. body matter
3. appendices
4. back matter

The document divisions are defined with:

```
\startfrontmatter ... \stopfrontmatter
\startbodymatter ... \stopbodymatter
\startappendices ... \stopappendices
\startbackmatter ... \stopbackmatter
```

The chapters in your book can be divided over these divisions.

```
\starttext
\startstandardmakeup
  \midaligned{From Hasselt to America}
  \midaligned{by}
  \midaligned{J. Jonker and C. van Marle}
\stopstandardmakeup
\startfrontmatter
  \title{Preface}
```



² Here we try to avoid the word *section*.





```

\chapter{Introduction}
\stopfrontmatter
\startbodymatter
\chapter{The Rensselaer family}
\chapter{The Lansing family}
\chapter{The Cuyler family}
\stopbodymatter
\startappendices
\chapter{Photos}
\stopappendices
\stoptext

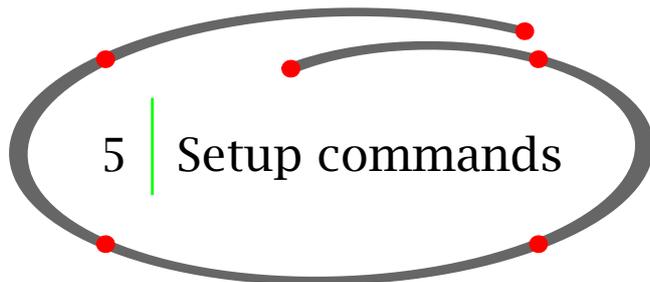
```

In the front matter as well as back matter the command `\chapter` produces an un-numbered header in the table of contents. The front matter is mostly used for the table of contents, the list of figures and tables, the preface, the acknowledgements etc. It often comes with a roman page numbering.

The appendices division is used for (indeed) appendices. Headers may be typeset in a different way; for example, `\chapter` may be numbered alphabetically.

The style of each document division can be set up with:

```
\setupsectionblock
```



Global commands are placed in the setup area of your input file, before `\starttext`. In appendix A there is a complete overview of the available commands and their parameters.

The set up commands all have the same structure. They look something like:

```
\setupparagraphs
```

Heads

A set up command consists of a more or less logical name and a number of bracket pairs. Bracket pairs may be optional and in that case the `[]` are typeset slanted `]`. In the definition the bracket pairs may contain:

```
\setupcommand[.1.][.2.][...=...]
```

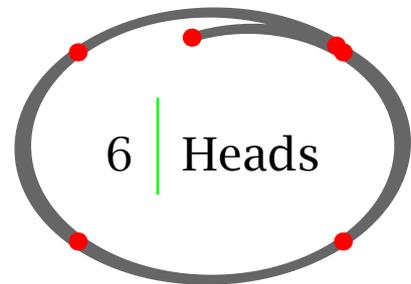
The commas indicate that a list of parameters can be enclosed. In the options list following the definition, the `.1.` and `.2.` show the possible options that can be set in the first and second bracket pair respectively. The parameters and their possible values are placed in the third bracket pair.

The default options and parameter values are underlined.

Furthermore you will notice that some values are typeset in a slanted way: *section*, *name*, *dimension*, *number*, *command* and *text*. This indicates that you can set the value yourself.

<i>section</i>	a section name like chapter, section, subsection etc.
<i>name</i>	an identifier (logical name)
<i>dimension</i>	a dimension with a unit in cm, pt, em, ex, sp or in
<i>number</i>	an integer
<i>command</i>	a command
<i>text</i>	text

In the Quick Reference manual you can find a complete overview of the commands and their parameters.



The structure of a document is determined by its chapter and section titles. These titles are created with the commands shown in table 6.1:

Numbered header	Unnumbered header
<code>\chapter</code>	<code>\title</code>
<code>\section</code>	<code>\subject</code>
<code>\subsection</code>	<code>\subsubject</code>
<code>\subsubsection</code>	<code>\subsubsubject</code>
...	...

Table 6.1 Headers.

6

`\chapter``\section``\subsection``\title``\subject``\subsubject`

These commands will produce a numbered or unnumbered title in a predefined fontsize and fonttype with some vertical spacing before and after the header.

The title commands can take several arguments, like in:

```
\title[hasselt by night]{Hasselt by night}
```

and

```
\title{Hasselt by night}
```

The bracket pair is optional and used for internal references. If you want to refer to this chapter you type for example `\at{page}[hasselt by night]`.

For a more structured way to define chapters and sections you can use the more preferred `\start ... \stop` construction.

In that case the definition looks like this:

```
\starttitle[reference="hasselt by night",title="Hasselt by night"]
...
\stoptitle
```

Numbered header	Un-numbered header
<code>\start ... \stopchapter</code>	<code>\start ... \stoptitle</code>
<code>\start ... \stopsection</code>	<code>\start ... \stopsubject</code>
<code>\start ... \stopsubsection</code>	<code>\start ... \stopsubsubject</code>
<code>\start ... \stopsubsubsection</code>	<code>\start ... \stopsubsubsubject</code>
...	...

Table 6.2 Structured headers.

Of course the chapter and section titles can be set to your own preferences and you can even define your own sections. This is done with the `\setuphead` and `\definehead` command.

```
\definehead
```

```
\setuphead
```

```
\definehead
  [myhead]
  [section]

\setuphead
  [myhead]
  [numberstyle=bold,
   textstyle=bold,
   before=\hairline\blank,
   after=\nowhitespace\hairline]

\myhead[headlines]{Hasselt makes headlines}
```

A new header `\myhead` is defined and it inherits the properties of `\section`. It would look something like this:

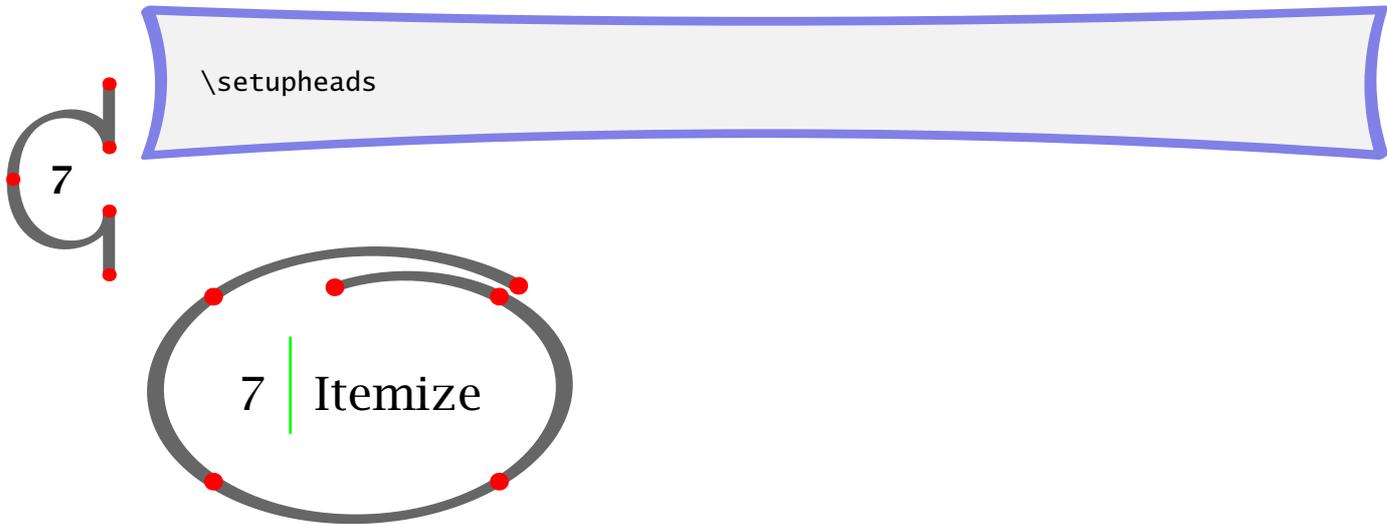
6.1 Hasselt makes headlines

There is one other command you should know now, and that is `\setupheads`. You can use this command to set up the numbering of the numbered chapters and sections. If you type:

```
\setupheads
  [alternative=inmargin,
   separator=--]
```

all numbers will appear in the margin. Section 1.1 would look like 1-1.

Commands like `\setupheads` are typed in the set up area of your input file.



One way of structuring your information is by way of enumeration or summing up items. The `itemize` command looks like this:

For example:

```
\startitemize[R,packed,broad]
\item Hasselt was founded in the 14th century.
\item Hasselt is known as a so called Hanze town.
\item Hasselt's name stems from a tree.
\stopitemize
```

Within the `\start ... \stopitemize` pair you start a new item with `\item`. The space after `\item` is required. In the example above R specifies Roman numbering and `packed` keeps line spacing to a minimum. The parameter `broad` takes care of the spacing between item separator and item. The example would produce:

- I. Hasselt was founded in the 14th century.
- II. Hasselt is known as a so called Hanze town.
- III. Hasselt's name stems from a tree.

Items can be defined in a more structured way:

```
\startitemize[R,packed,broad]
\startitem Hasselt was founded in the 14th century. \stopitem
\startitem Hasselt is known as a so called Hanze town. \stopitem
\startitem Hasselt's name stems from a tree. \stopitem
\stopitemize
```

The bracket pair contains information on item separators and local set up variables.

Argument	Item separator symbol
1	•
2	—
3	★
:	:
n	1 2 3 4 ...
a	a b c d ...
A	A B C D ...
r	i ii iii iv ...
R	I II III IV ...

Table 7.1 Item separators in itemize.

You can also define your own item separator by means of `\definesymbol`. For example if you try this:

```
\definesymbol[5][\clubsuit$]
\startitemize[5,packed]
\item Hasselt was built on a riverdune.
\item Hasselt lies at the crossing of two rivers.
\stopitemize
```

You will get:

- ♣ Hasselt was built on a riverdune.
- ♣ Hasselt lies at the crossing of two rivers.

If you want to have a sort of head within an enumeration you should use `\head` instead of `\item`.

Hasselt lies in the province of Overijssel and there are a number of customs that are typical of this province.

```
\startitemize
```

```
\head kraamschudden \hfill (child welcoming)
```

When a child is born the neighbours come to visit the new parents. The women come to admire the baby and the men come to judge the baby (if it is a boy) on other aspects.

The neighbours will bring a `{\em krentenwegge}` along. A `krentenwegge` is a loaf of currant bread of about `1 \unit{Meter}` long. Of course the birth is celebrated with `{\em jenever}`.

```
\head nabuurschap (naberschap) \hfill (neighbourship)
```

Smaller communities used to be very dependent on the cooperation among the members for their well being. Members of the `{\em nabuurschap}` helped each other in difficult times during harvest times, funerals or any hardship that fell upon

the community.

```
\head Abraham \& Sarah \hfill (identical)
```

When people turn 50 in Hasselt it is said that they see Abraham or Sarah. The custom is to give these people a `{\em speculaas}` Abraham or a Sarah. Speculaas is a kind of hard spiced biscuit.

```
\stopitemize
```

The `\head` can be set up with `\setupitemize`. In case of a page breaking a `\head` will appear on a new page. (The `\unit{Meter}` command is explained in chapter 10.)

The example of old customs will look like this:

Hasselt lies in the province of Overijssel and there are a number of customs that are typical of this province.

- kraamschudden (child welcoming)

When a child is born the neighbours come to visit the new parents. The women come to admire the baby and the men come to judge the baby (if it is a boy) on other aspects. The neighbours will bring a *krentenwegge* along. A *krentenwegge* is a loaf of currant bread of about 1 m long. Of course the birth is celebrated with *jenever*.

- nabuurschap (naberschap) (neighbourship)

Smaller communities used to be very dependent on the cooperation among the members for their well being. Members of the *nabuurschap* helped each other in difficult times during harvest times, funerals or any hardship that fell upon the community.

- Abraham & Sarah (identical)

When people turn 50 in Hasselt it is said that they see Abraham or Sarah. The custom is to give these people a *speculaas* Abraham or a Sarah. Speculaas is a kind of hard spiced biscuit.

The set up parameters of `itemize` are described in table 7.2.

Set up	Meaning
standard	standard (global) set up
packed	no vertical spacing between items
serried	no horizontal spacing between separator and text
joinedup	no vertical spacing before and after itemize
broad	horizontal spacing between separator and text
inmargin	place separator in margin
atmargin	place separator on margin
stopper	place full stop after separator
columns	put items in columns
intro	prevent page breaking after introduction line
continue	continue numbering or lettering

Table 7.2 Set up parameters in `itemize`.

Itemize

You can use the set up parameters in `\startitemize`, but for reasons of consistency you can make them valid for the complete document with `\setupitemize`.

The parameter `columns` is used in conjunction with a (written) number. If you type this:

```
\startitemize[n,columns,four]
\item Achter 't Werk
.
.
.
\item Justitiebastion
\stopitemize
```



You will get:

- | | | | |
|------------------------|------------------------|--------------------------|------------------|
| 1. Achter 't Werk | 5. Eiland | 9. Hoogstraat | 13. Kalverstraat |
| 2. Baangracht | 6. Gasthuis-
straat | 10. Julianakade | 14. Kastanjelaan |
| 3. Brouwers-
gracht | 7. Heerengracht | 11. Justitiebas-
tion | 15. Keppelstraat |
| 4. Eikenlaan | 8. Hofstraat | 12. Kaai | |

Sometimes you want to continue the enumeration after a short intermezzo. Then you type for example `\startitemize[continue]` and numbering will continue and all other preferences are kept.

- | | | | |
|-------------------|-------------------------|-------------------------|---------------------------|
| 16. Markt | 21. Rosmolen-
straat | 24. Vicariehof | 28. Ziekenhuis-
straat |
| 17. Meestersteeg | 22. Royenplein | 25. Vissteeg | |
| 18. Prinsengracht | 23. Van Nahui-
jsweg | 26. Watersteeg | |
| 19. Raamstraat | | 27. Wilhelmi-
nalaan | |
| 20. Ridderstraat | | | |

The parameter `broad` enlarges the horizontal space between item separator and itemtext.

An itemize within an itemize is automatically typeset in a correct way. For example if you type:

In the Netherlands the cities can determine the height of a number of taxes. So the cost of living can differ from town to town. There are differences of up to 50% in taxes such as:

```
\setupitemize[2][width=5em]
\startitemize[n]
\item[estate tax] real estate tax
```

The real estate tax is divided into two components:

```
\startitemize[a,packed]
\item the ownership tax
\item the tenant tax
\stopitemize
```

If the real estate has no tenant the owner pays both components.

```
\item dog licence fee
```

The owner of one or more dogs pays a fee. When a dog has died

or been sold the owner has to inform city hall.

`\stopitemize`

then the horizontal space between item separator and text at the second level of itemizing is set with `\setupitemize[2][width=5em]`.

The example will look like this:

In the Netherlands the cities can determine the height of a number of taxes. So the cost of living can differ from town to town. There are differences of up to 50% in taxes such as:

1. real estate tax

The real estate tax is divided into two components:

- a. the ownership tax
- b. the tenant tax

If the real estate has no tenant the owner pays both components.

2. dog licence fee

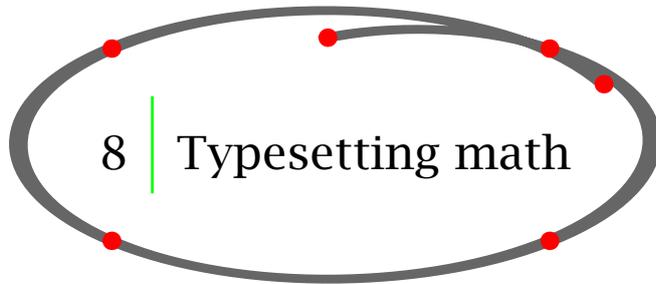
The owner of one or more dogs pays a fee. When a dog has died or been sold the owner has to inform city hall.

You can refer to an item if you give it a label (see `\item[estate tax]`). If you then type:

`\in{In item}[estate tax]` we discussed one of the income sources of Hasselt.

You'll get a reference to that item:

In item 1 we discussed one of the income sources of Hasselt.



8.1 Introduction

\TeX is *the* typesetting program for math. However, this is not the extensive chapter on typesetting math you might expect. We advise you to do some further reading on typesetting formulas in \TeX . See for example:³

- *The \TeX Book* by D.E. Knuth
- *The Beginners Book of \TeX* by S. Levy and R. Seroul

³ In this introduction on typesetting math we relied on the booklet *\TeX niques* by Arthur Samuel.

8.2 Typesetting math

Normally different conventions are applied for typesetting normal text and math text. These conventions are ‘known’ by $\text{T}_{\text{E}}\text{X}$ and applied accordingly when generating a document. We can rely on $\text{T}_{\text{E}}\text{X}$ for delivering high quality math output.

A number of conventions for math are:

1. Characters are typeset in *math italic* (don’t confuse this with the normal *italic characters* in a font).
2. Symbols like Greek characters (α , χ) and math symbols (\leq , \geq , \in) are used.
3. Spacing will differ from normal spacing.
4. Math expressions have a different alignment than that of the running text.
5. The sub and superscripts are downsized automatically, like in a_c^b .
6. Certain symbols have different appearances in the inline and display mode.

When typesetting math you have to work in the so called math mode in which math expressions can be defined by means of plain $\text{T}_{\text{E}}\text{X}$ -commands.

Math mode has two alternatives: text mode and display mode. Math in text mode is activated by $\$$ and $\$$, while display mode is activated by $\$\$$ and $\$\$$. In $\text{CON}_{\text{T}}\text{E}_{\text{X}}\text{T}$ however, display mode is activated with the $\backslash\text{start} \dots \backslash\text{stopformula}$ command pair to have more grip on vertical spacing around the formula.

The municipality of Hasselt covers an area of 42,05 $\backslash\text{unit}\{\text{Square Kilo Meter}\}$. Now, if you consider a circular area of this size with the market place of Hasselt as the center point M you can calculate its diameter with $\$\{\{1\}\over{4}\} \backslash\pi r^2\$$.

This will become:

The municipality of Hasselt covers an area of 42,05 km^2 . Now, if you consider a circular area of this size with the market place of Hasselt as the center point M you can calculate its diameter with $\frac{1}{4}\pi r^2$.

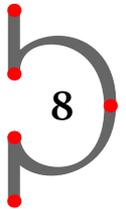
The many $\{\}$ (grouping) in $\frac{1}{4}\pi r^2$ are essential for separating operations in the expression. If you omit the outer curly braces like this: $\$\{1\}\over{4} \backslash\pi r^2\$$, you would get a non desired result: $\frac{1}{4\pi r^2}$.

The letters and numbers are typeset in three different sizes: text size $a + b$, script size $a+b$ and scriptscript size $a+b$. These can be influenced by the commands $\backslash\text{scriptstyle}$ and $\backslash\text{scriptscriptstyle}$.

Symbols like \int and \sum will have a different form in text and display mode. If we type $\$\sum_{n=1}^m\$$ or $\$\int_{-\infty}^{+\infty}\$$ we will get $\sum_{n=1}^m$ and $\int_{-\infty}^{+\infty}$. But when you type:

```
\startformula
  \sum_{n=1}^m \quad {\rm and} \quad \int_{-\infty}^{+\infty}
\stopformula
```

to get displaymode you get:



$$\sum_{n=1}^m \quad \text{and} \quad \int_{-\infty}^{+\infty}$$

With the commands `\nolimits` and `\limits` you can influence the appearances of `\sum` and `\int`:

```
\startformula
\sum_{n=1}^m\nolimits
\quad {\rm and} \quad
\int_{-\infty}^{+\infty}\limits
\stopformula
```

which will result in:

$$\sum_{n=1}^m \quad \text{and} \quad \int_{-\infty}^{+\infty}$$

For typesetting fractions there is the command `\over`. In `CONTEXT` you can use the alternative `\frac`. For $\frac{a}{1+b} + c$ we type for instance `$$\frac{a}{1+b}+c$`.

Other commands to put one thing above the other, are:

```
\atop   ${a} \atop {b}$       $\frac{a}{b}$ 
\choose ${n+1} \choose {k}$   $\binom{n+1}{k}$ 
\brack  ${m} \brack {n}$      $\left[ \begin{matrix} m \\ n \end{matrix} \right]$ 
\brace  ${m} \brace {n-1}$    $\left\{ \begin{matrix} m \\ n-1 \end{matrix} \right\}$ 
```

`TEX` can enlarge delimiters like `()` and `{}` automatically if the left and right delimiter is preceded by the commands `\left` and `\right` respectively. If you type:

```
\startformula
1+\left(\frac{1}{1-x^{x-2}}\right)^3
\stopformula
```

you will get:

$$1 + \left(\frac{1}{1 - x^{x-2}} \right)^3$$

Sub and superscripts are invoked by `'_'` and `'^'`. They have effect on the next first character so grouping with `{}` is necessary in case of multi character sub and superscripts.

In certain situations the delimiters can be preceded by `\bigl`, `\Bigl`, `\bigr` and `\Bigr` and their right counterparts. Even bigger delimiters can be produced by placing `\left` and `\right` in a `\vbox` construction. When we type a senseless expression like:

```
\startformula
\left(\vbox to 16pt{x^{2^{2^{2^2}}}}\right)
\stopformula
```

we get:

$$\left(x^{2^{2^2}}\right)$$

In display mode the following delimiters will work in the automatic enlargement mechanism:

<code>\lfloor</code>		<code>\lceil</code>	<	<code>\vert</code>		<code>\downarrow</code>	↓
<code>\rfloor</code>		<code>\rceil</code>	>	<code>\Vert</code>		<code>\Downarrow</code>	⇓
<code>\lceil</code>		/	/	<code>\uparrow</code>	↑	<code>\updownarrow</code>	↕
<code>\rceil</code>		<code>\backslash</code>	\	<code>\Uparrow</code>	↑	<code>\Updownarrow</code>	↕

In display mode we should typeset only one fraction and otherwise switch to the a/b notation. To get:

$$a_0 + \frac{a}{a_1 + \frac{1}{a_2}}$$

we will not type:

```
\startformula
  a_0+{\frac{a}{a_1+\frac{1}{a_2}}}
\stopformula
```

but prefer:

```
\startformula
  a_0 + {\frac{a}{a_1 + 1/a_2}}
\stopformula
```

to obtain:

$$a_0 + \frac{a}{a_1 + 1/a_2}$$

In addition we could also use the command `\displaystyle`. If we would type:

$$a_0 + \frac{a}{a_1 + \frac{1}{a_2}}$$

we will get:

$$a_0 + \frac{a}{a_1 + \frac{1}{a_2}}$$

Below we demonstrate the commands `\matrix`, `\pmatrix`, `\ldots`, `\cdots` and `\cases` without any further explanation.

```
\startformula
A=\left(\matrix{x-\lambda & 1 & & 0 \\
0 & x-\lambda & & 1 \\
0 & & 0 & x-\lambda} \right)
\stopformula
```

$$A = \begin{pmatrix} x - \lambda & 1 & 0 \\ 0 & x - \lambda & 1 \\ 0 & 0 & x - \lambda \end{pmatrix}$$

`\startformula`

```
A=\left|\matrix{x-\mu& 1 & 0 \\ 0 & x-\mu & 1 \\ 0 & 0 & x-\mu}\right|
```

`\stopformula`

$$A = \begin{vmatrix} x - \mu & 1 & 0 \\ 0 & x - \mu & 1 \\ 0 & 0 & x - \mu \end{vmatrix}$$

`\startformula`

```
A=\pmatrix{a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn}}
```

`\stopformula`

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}$$

`\startformula`

```
A=\pmatrix{a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn}}
```

`\stopformula`

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}$$

`\startformula`

```
|x|=\cases{x, & if $x\geq 0$; \\ -x, & otherwise}
```

`\stopformula`

$$|x| = \begin{cases} x, & \text{if } x \geq 0; \\ -x, & \text{otherwise} \end{cases}$$



To typeset normal text in a math expression we have to consider the following. First a space is not typeset in math mode so we have to enforce one with `\` (backslash). Second we have to indicate a font switch, because the text should not appear in *math italic* but in the actual font. So in `CONTEXT` we have to type:

```
\startformula
  x^3+{\tf lower\ order\ terms}
\stopformula
```

to get:

$$x^3 + \text{lower order terms}$$

The math functions like `sin` and `tan` that have to be typeset in the actual font are predefined functions in `TEX`:

```
\arccos \cos \csc \exp \ker \limsup \min \sinh
\arcsin \cosh \deg \gcd \lg \ln \Pr \sup
\arctan \cot \det \hom \lim \log \sec \tan
\arg \coth \dim \inf \liminf \max \sin \tanh
```

If we type the sinus or limit function:

```
\startformula
  \sin 2\theta=2\sin\theta\cos\theta
  \quad {\tf or} \quad
  \lim_{x\to0}{\frac{\sin x}{x}}=1
\stopformula
```

we get:

$$\sin 2\theta = 2 \sin \theta \cos \theta \quad \text{or} \quad \lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$$

Alignment in math expressions may need special attention. In multi line expressions we sometimes need alignment at the '=' sign. This is done by the command `\eqalign`. If we type:

```
\startformula
  \eqalign{
    ax^2+bx+c &= 0 & \cr
    x &=& \frac{-b \pm \sqrt{b^2-4ac}}{2a} & \cr}
\stopformula
```

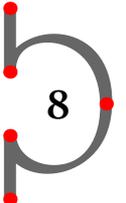
we get:

$$ax^2 + bx + c = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Sometimes alignment at more than one location is wanted. Watch the second line in the next example and see how it is defined:

```
\startformula
```



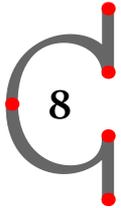
```

\eqalign{
ax+bx+\cdots+yx+zx & = x(a +b+ \cdots \cr
& \phantom{= x(a~}+y+z) \cr
& = y \cr}
\stopformula

```

This results in:

$$\begin{aligned}
 ax + bx + \cdots + yx + zx &= x(a + b + \cdots \\
 &+ y + z) \\
 &= y
 \end{aligned}$$



8

Next to the command `\phantom` there are `\hphantom` without height and depth and `\vphantom` without width.

You can rely on \TeX for spacing within a math expression. In some situations, however you may want to influence spacing. This is done by:

```

\! -\frac{1}{6}\quad
\, \frac{1}{6}\quad
\> \frac{2}{9}\quad
\; \frac{5}{18}\quad

```

These ‘spaces’ are related to `\quad` that stands for the width of the capital ‘M’.

The use of the command `\prime` speaks for itself. For example if you would want $y'_1 + y''_2$ you should type `\$y_1^\prime+y_2^\{\prime\prime}\$`.

An expression like $\sqrt[3]{x^2 + y^2}$ is obtained by `\$ \root 3 \ of \{x^2+y^2\}\$`.

At the end of this section we point to the command `\mathstrut` which we can use to enforce consistency, for example within the root symbol. With `\$ \sqrt{\mathstrut a} + \sqrt{\mathstrut d} + \sqrt{\mathstrut y}\$` we will get $\sqrt{a} + \sqrt{d} + \sqrt{y}$ in stead of $\sqrt{a} + \sqrt{d} + \sqrt{y}$.

See appendix E for a complete overview of math commands.

8.3 Placing formulas

You can typeset numbered formulas with:

```
\placeformula
```

```
\startFORMULA
```

Two examples:

```
\placeformula[formula:aformula]
```

```
\startformula
```

```
y=x^2
```

```
\stopformula
```

```
\placeformula
```

```
\startformula
```

```
\int_0^1 x^2 dx
```

```
\stopformula
```

$$y = x^2 \tag{8.1}$$

$$\int_0^1 x^2 dx \tag{8.2}$$

The command `\placeformula` handles spacing around the formulas and the numbering. The bracket pair is optional and is used for referencing and to switch numbering on and off.

$$y = x^2 \tag{8.3}$$

$$y = x^3 \tag{8.4}$$

$$y = x^4 \tag{8.5}$$

Formula 8.4 was typed like this:

```
\placeformula[middle one]
```

```
\startformula
```

```
y=x^3
```

```
\stopformula
```

The label `[middle one]` is used for referring to this formula. Such a reference is made with `\in{formula}[middle one]`.

If no numbering is required you type:

```
\placeformula[-]
```

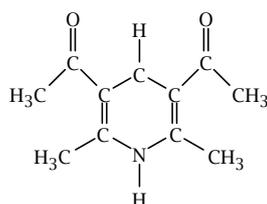
Numbering of formulas is set up with `\setupnumbering`. In this manual numbering is set up with `\setupnumbering[way=bychapter]`. This means that the chapter number precedes the formula number and numbering is reset with each new chapter. For reasons of consistency the tables, figures, intermezzi etc. are numbered in the same way. Therefore you use `\setupnumbering` in the set up area of your input file.

Formulas can be set up with:

9 | Chemical stuff

9

Chemical structures may look very impressive.



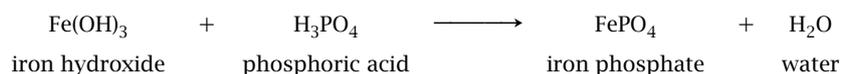
Compound A

CONTEX_T relies on METAPOST to draw these kind of chemical structures. Although these chemical structures are defined with only two or three commands, it takes some practice to get the right results. This is how the input looks:

```
\startchemical [scale=small, width=fit, top=3000, bottom=3000]
\chemical [SIX, SB2356, DB14, Z2346, SR3, RZ3, SR6, -RZ6, +RZ6]
[C, N, C, C, H, H]
\chemical [PB:Z1, ONE, Z0, MOV8, Z0, SB24, DB7, Z27, PE] [C, C, CH_3, O]
\chemical [PB:Z5, ONE, Z0, MOV6, Z0, SB24, DB7, Z47, PE] [C, C, H_3C, O]
\chemical [SR24, RZ24] [CH_3, H_3C]
\bottomtext{Compound A}
\stopchemical
```

Chemical reactions can be typeset within a paragraph or as a display formula with the `\inlinechemical` and `\startchemicalformula` commands:

One of the steps in the Hasselt canal water treatment is the removal of phosphate by means of a chemical reaction with iron:



The FePO_4 is a solid and precipitates in water. It is filtered and re-used as a fertilizer resource.

This is defined by:

```
\definefloat
[chemicalformula]
[chemicalformulas]
```

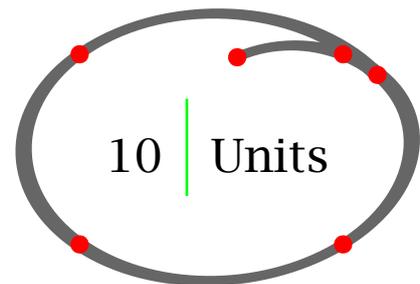
Units

One of the steps in the Hasselt canal water treatment is the removal of phosphate by means of a chemical reaction with iron:

```
\placechemicalformula[none] [] {}
  {\startchemicalformula
    \chemical{Fe(OH)_3}{iron hydroxide}
    \chemical{PLUS}
    \chemical{H_3PO_4}{phosphoric acid}
    \chemical{GIVES}{\hphantom{whatever}}
    \chemical{FePO_4}{iron phosphate}
    \chemical{PLUS}
    \chemical{H_2O}{water}
  \stopchemicalformula}
```

The `\inlinechemical{FePO_4}` is a solid and precipitates in water. It is filtered and re-used as a fertilizer resource.

The use of the chemical commands is described in the PPCHTeX Manual and the example manual Chemical Formulas in CONT_{TeX}T.



To force yourself to use dimensions and units consistently throughout your document you can use the `\unit` command. Let's give a few examples:

```
\unit{meter per square meter}
\unit{cubic meter per sec}
\unit{square milli meter per inch}
\unit{centi liter per sec}
\unit{meter inverse sec}
\unit{newton per square inch}
\unit{newton times meter per square sec}
```

It looks like a lot of typing but it does guarantee a consistent use of units. The command `\unit` also prevents linebreaking between number and unit. The examples above come out as:

```
m/m2
m3/s
mm2/inch
cl/s
m·s-1
```



N/inch²
N·m/s²

You can add your own units with:
and set them up with:

In the example below you can see some new units and the non-consistent use of km.

```
\registerunit[unit][inhab=inhabitants] \setupunittext[inhabitants=inh]
\registerunit[unit][north=north]      \setupunittext[north= N]
\registerunit[unit][east=east]        \setupunittext[east= E]
```

Hasselt is part of the municipality of Zwartewaterland (coordinates `\unit {52 degrees 35 arcminute north}`, `\unit {6 degrees 5 arcminute east}`). Its area is about `\unit {88 square kilometer}` (land `\unit {83 square kilom}` and water `\unit{5 square km}`). As of 1st Augustus 2013 the population is 22.201 that is `\unit {268 inhab per square kilo meter}`).

This results in:

Hasselt is part of the municipality of Zwartewaterland (coordinates 52° 35´ N, 6° 5´ E). Its area is about 88 km² (land 83 km² and water 5 km²). As of 1st Augustus 2013 the population is 22.201 that is 268 inh/km²).

The `\unit` command also allows you to align rows of units in a column.

When you type:

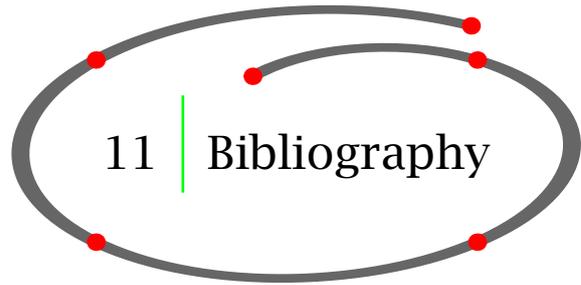
```
\bTABLE
\bTR \bTD \bf Street          \eTD \bTD \bf Length          \eTD \eTR
\bTR \bTD Ridderstraat       \eTD \bTD \unit{_,160 meter} \eTD \eTR
\bTR \bTD Prinsengracht      \eTD \bTD \unit{_,240 meter} \eTD \eTR
\bTR \bTD Kalverstraat       \eTD \bTD \unit{_,_60 meter} \eTD \eTR
\bTR \bTD H.A.W. van de Vechtlaan \eTD \bTD \unit{1,250 meter} \eTD \eTR
\bTR \bTD Meestersteeg       \eTD \bTD \unit{_,_45 meter} \eTD \eTR
\eTABLE
```

It will generate a well aligned second column:

Street	Length
Ridderstraat	160 m
Prinsengracht	240 m
Kalverstraat	60 m
H.A.W. van de Vechtlaan	1,250 m
Meestersteeg	45 m

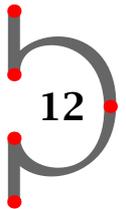
Please refer to the manual *Units* for more information and details.





CONTEX support the BIBTEX way of managing article and book references. The data is stored in a .bib file. A data entry in a BIBTEX data file could be:

```
@INBOOK{book01,
  author = "Jonker, J.",
  title = "From Hasselt to America",
  publisher = "Bookplan Publishers",
  year = "2012",
  chapter = "1.2",
}
```



After loading the database with `\setupbibtex[database=hasseltbook]` the following command is available:

```
Please refer to \cite[book01] for more information on famous people
that were born in Hasselt.
```

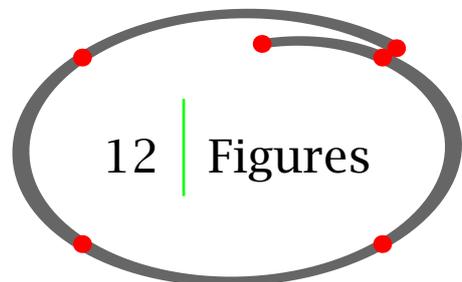
Which would produce:

```
Please refer to Jonker (2012) for more information on famous people that were born in Hasselt.
```

In an appendix you can place the complete book list with:

```
\placepublications[criterium=text]
```

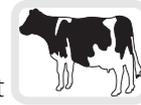
At this moment (2013 - 2014) the bibliography mechanism is being completely overhauled, so please visit the CONTEX WIKI and the Pragma ADE website regularly for information.



Images can be placed in your document with the command `\externalfigure`.

```
\externalfigure
[cow.pdf]
```

```
[width=.1\textwidth,
frame=on,
framecolor=gray,
frameoffset=3pt,
rulethickness=3pt,
framecorner=round]
```



12

Such an image will be placed on the location where you defined it and can have some strange effects on the surrounding white space. By the way, the cow image is always available for `CONTEX`T users which is very convenient when you are testing the figure related commands.

You can use the command `\placefigure` to influence the positioning of images in your document.

```
\placefigure
[] [fig:church]
{Stephanus Church.}
{\externalfigure[ma-cb-24] [width=.4\textwidth]}
```

After processing this will come out as figure 12.1 at the first available location.

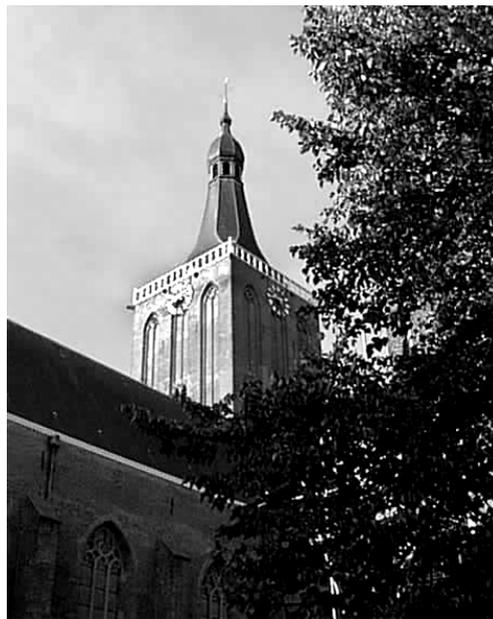


Figure 12.1 Stephanus Church.

The command `\placefigure` handles numbering and vertical spacing before and after your figure. Furthermore this command initializes a float mechanism, which means that `CONTEX`T looks whether there is enough space for your figure on the page. If not, the figure will be placed at another location and the text carries on, while the figure floats in your document until the optimal location is found. You can influence this mechanism within the first bracket pair.

The command `\placefigure` is a predefined example of:

`\placeFLOAT`

A number of basic options is described in table 12.1.

Option	Meaning
here	put figure at this location if possible
force	force figure placement here
page	put figure on its own page
top	put the figure at the top of the page
bottom	put the figure at the bottom of the page
left	place figure at the left margin
right	place figure at the right margin
margin	place figure in the margin
none	set no caption

Table 12.1 Options in `\placefigure`.

The second bracket pair is used for cross-referencing. You can refer to this particular figure by typing:

```
\in{figure}[fig:church]
```

The first brace pair is used for the caption. You can type any text you want. The figure labels are set up with `\setupcaptions` and the numbering is (re)set by `\setupnumbering` (see paragraph 40.5).

The second brace pair is used for defining the figure and addressing the file names of external figures.

In the next example you see how `Hasselt` is defined within the last brace pair to show you the function of `\placefigure`.

```
\placefigure
  {The boundaries of Hasselt.}
  {\framed{\tfd Hasselt}}
```

This will produce:

Hasselt

Figure 12.2 The boundaries of Hasselt.

However, your images are often created using programs like Illustrator and photos are — after scanning — improved in packages like PhotoShop. Then the images are available as files.

CONTEX supports image file types like JPG, PNG and (pages from) PDF files as well as METAPOST output (MPS files). Users normally can trust CONTEX to find the best possible file type.

In figure 12.3 you see a photo and a graphic combined into one figure.

12



a bitmap picture



a vector graphic

Figure 12.3 The Hasselt Canals.

You can produce this figure by typing something like:

```
\placefigure
  [here,force]
  [fig:canals]
  {The Hasselt Canals.}
  {\startcombination[2*1]
    {\externalfigure[ma-cb-03][width=.4\textwidth]}{a bitmap picture}
    {\externalfigure[ma-cb-00][width=.4\textwidth]}{a vector graphic}
  \stopcombination}
```

In this figure two pictures are combined with:

```
\startcombination
```

The `\start ... \stopcombination` pair is used for combining two pictures in one figure. You can type the number of pictures within the bracket pair. If you want to display one picture below the other you would have typed `[1*2]`. You can imagine what happens when you combine 6 pictures as `[3*2]` (`[rows*columns]`).

The examples shown above are enough for creating illustrated documents. Sometimes however you want a more integrated layout of the picture and the text. For that purpose you can use `\start ... \stopfiguretext` command pair.

The effect of:

```
\startfiguretext
  [left,none]
  [fig:citizens]
```

```
{}
```

```
{\externalfigure[ma-cb-18][width=.5\makeupwidth]}
```

Hasselt has always had a varying number of citizens due to economic events. For example the Dedemsvaart was dug around 1810. This canal runs through Hasselt and therefore trade flourished. This led to a population growth of almost 40% within 10~years. Nowadays the Dedemsvaart has no commercial value anymore and the canals have become a tourist attraction. But reminders of these prosperous times can be found everywhere.

```
\stopfiguretext
```

is shown in the figure below.



Hasselt has always had a varying number of citizens due to economic events. For example the Dedemsvaart was dug around 1810. This canal runs through Hasselt and therefore trade flourished. This led to a population growth of almost 40% within 10 years. Nowadays the Dedemsvaart has no commercial value anymore and the canals have become a tourist attraction. But reminders of these prosperous times can be found everywhere.

As you have seen you in the examples above you can summon a figure with the command:

```
\externalfigure
```



The command `\externalfigure` has two bracket pairs. The first is used for the exact file name without extension, the second for file formats and dimensions. It is not difficult to guess what happens if you type:

```
\inmargin
  {\externalfigure
    [ma-cb-23]
    [width=.7\marginwidth]}
```

You can set up the layout of figures with:

```
\setupfloats
```

You can set up the numbering and the labels with:

```
\setupcaptions
```

These commands are typed in the set up area of your input file and have a global effect on all floating blocks.

```
\setupfloat
 [figure]
 [default=right,
  spacebefore=none]
\setupcaptions
 [location=bottom,
  style=boldslanted]
\placefigure
 {A characteristic view on Hasselt.}
 {\externalfigure[ma-cb-12] [width=6cm]}
```

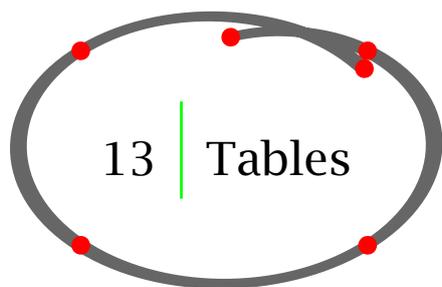
For figure management there are commands like `\setupexternalfigure`.

Please refer to the `CONTEXT WIKI` for practical applications of these commands.

If you want to work with a XML based figure database please see the Figures manual.



Figure 12.4 *A characteristic view on Hasselt.*



There are a number of ways to define a table:

- the `\start ... \stoptable` mechanism, based on the work of M. Wichura
- the `\bTABLE ... \eTABLE` mechanism (natural tables)
- the `\start ... \stopxtable` mechanism (extreme tables)

In the next sections we describe the principles of the three table mechanisms.

13.1 Simple tables

For defining the table you use:

```
\starttable
```

The definition of a table could look something like this:

```
\placetable
[here]
[tab:ships]
{Ships that moored at Hasselt.}
{\starttable[|c|c|]
\HL
\NC \bf Year \NC \bf Number of ships \NC\SR
\HL
\NC 1645 \NC 450 \NC\FR
\NC 1671 \NC 480 \NC\MR
\NC 1676 \NC 500 \NC\MR
\NC 1695 \NC 930 \NC\LR
\HL
\stoptable}
```

This table is typeset as table 13.1.

Year	Number of ships
1645	450
1671	480
1676	500
1695	930

Table 13.1 Ships that moored at Hasselt.

Although this table mechanism is still available and supported in `CONTEXT` it is better to use one of the other mechanisms.

13.2 Natural tables

The natural table mechanism (`\bTABLE ... \eTABLE`) is developed for more complex tables and has features of the general interface of `CONTEXT`.

```
\placetable
[here,force]
[tab:votedivision]
{Division of votes over political parties.}
{\bTABLE[align=middle,offset=4pt]
\bTABLEhead
\bTR[width=6cm] \bTD [nc=5] Elections City Council \eTD \eTR
\eTABLEhead}
```


Tables

```

\begin{table}
\begin{tablehead}
\begin{tabletr}
\begin{tabletd}[nc=5] Elections City Council \end{tabletd} \end{tabletr}
\begin{tabletr}
\begin{tabletd}[nr=2] Party \end{tabletd} \begin{tabletd}[nc=3] Districts \end{tabletd} \begin{tabletd}[nr=2] Total \end{tabletd} \end{tabletr}
\begin{tabletr}
\begin{tabletd} \begin{tabletd} 1 \end{tabletd} \begin{tabletd} 2 \end{tabletd} \begin{tabletd} 3 \end{tabletd} \end{tabletr}
\end{tablehead}
\begin{tablebody}
\begin{tabletr}
\begin{tabletd} PvdA \end{tabletd} \begin{tabletd} 351 \end{tabletd} \begin{tabletd} 433 \end{tabletd} \begin{tabletd} 459 \end{tabletd} \begin{tabletd} 1243 \end{tabletd} \end{tabletr}
\begin{tabletr}
\begin{tabletd} CDA \end{tabletd} \begin{tabletd} 346 \end{tabletd} \begin{tabletd} 350 \end{tabletd} \begin{tabletd} 285 \end{tabletd} \begin{tabletd} ~981 \end{tabletd} \end{tabletr}
\begin{tabletr}
\begin{tabletd} VVD \end{tabletd} \begin{tabletd} 140 \end{tabletd} \begin{tabletd} 113 \end{tabletd} \begin{tabletd} 132 \end{tabletd} \begin{tabletd} ~385 \end{tabletd} \end{tabletr}
\begin{tabletr}
\begin{tabletd} SGP \end{tabletd} \begin{tabletd} 348 \end{tabletd} \begin{tabletd} 261 \end{tabletd} \begin{tabletd} 158 \end{tabletd} \begin{tabletd} ~767 \end{tabletd} \end{tabletr}
\begin{tabletr}
\begin{tabletd} GPV \end{tabletd} \begin{tabletd} 117 \end{tabletd} \begin{tabletd} 192 \end{tabletd} \begin{tabletd} 291 \end{tabletd} \begin{tabletd} ~600 \end{tabletd} \end{tabletr}
\end{tablebody}
\end{table}

```

The meaning of the `CONTEXT` commands are indicated in table 13.3.

Command	Meaning
<code>\begin{table} ... \end{table}</code>	begin end table
<code>\begin{tabletr} ... \end{tabletr}</code>	begin end row
<code>\begin{tabletd} ... \end{tabletd}</code>	begin end column
<code>\begin{tablehead} ... \end{tablehead}</code>	begin end tablehead
<code>\begin{tablebody} ... \end{tablebody}</code>	begin end tablebody
<code>\begin{tablefoot} ... \end{tablefoot}</code>	begin end tablefoot
<code>\setupTABLE</code>	table setup

Table 13.3 Commands to define natural tables.

You can find more information on this table mechanism on the `CONTEXT` WIKI and examples in the *Natural Tables* manual.

13.3 Extreme tables

For large tables that extend over a number of pages and where you want the table head repeated after each pagebreak `CONTEXT` has the *extreme table* mechanism.

```

\setupxtable[split=yes,header=repeat]
\setupxtable[offset=4pt]

\placetable
[ ]
[tab:wealthdecline]
{Decline of wealth through the ages.}
{\startxtable
\startxtablehead[align=middle,foregroundstyle=bold]
\startxrow
\startxcell[nx=6]
Decline of wealth in Dutch florine (Dfl)
\stopxcell
\stopxrow
\startxrow[foregroundstyle=bold]

```

Tables

```

\startxcell[width=1.2cm] Year \stopxcell
\startxcell 1.000--2.000 \stopxcell
\startxcell 2.000--3.000 \stopxcell
\startxcell 3.000--5.000 \stopxcell
\startxcell 5.000--10.000 \stopxcell
\startxcell over 10.000 \stopxcell
\stopxrow
\stopxtablehead
\startxtablenext
\startxrow
\startxcell[nx=6,align=middle,foregroundstyle=bold]
Decline of wealth in Dutch florine (Dfl) / Continued
\stopxcell
\stopxrow
\startxrow[foregroundstyle=bold]
\startxcell Year \stopxcell
\startxcell 1.000--2.000 \stopxcell
\startxcell 2.000--3.000 \stopxcell
\startxcell 3.000--5.000 \stopxcell
\startxcell 5.000--10.000 \stopxcell
\startxcell over 10.000 \stopxcell
\stopxrow
\stopxtablenext
\startxtablebody[align=middle]
\startxrow
\startxcell 1675 \stopxcell
\startxcell 22 \stopxcell
\startxcell ~7 \stopxcell
\startxcell ~5 \stopxcell
\startxcell ~4 \stopxcell
\startxcell ~5 \stopxcell
\stopxrow
\startxrow
\startxcell 1724 \stopxcell
\startxcell ~4 \stopxcell
\startxcell ~4 \stopxcell
\startxcell -- \stopxcell
\startxcell ~4 \stopxcell
\startxcell ~3 \stopxcell
\stopxrow
\startxrow
\startxcell 1750 \stopxcell
\startxcell 12 \stopxcell
\startxcell ~3 \stopxcell
\startxcell ~2 \stopxcell
\startxcell ~2 \stopxcell
\startxcell -- \stopxcell
\stopxrow
\startxrow
\startxcell 1808 \stopxcell
\startxcell ~9 \stopxcell
\startxcell ~2 \stopxcell
\startxcell -- \stopxcell
\startxcell -- \stopxcell
\startxcell -- \stopxcell

```

Tables

```
\stopxrow
\stopxtablebody
\stopxtable}
```

With the `\setupxtable` it is indicated that the table is allowed to split at a pagebreak and that the head should contain the content of the `\start ... \stopxtablenext`.

The result of this definition is shown in table 13.4.

Decline of wealth in Dutch florine (Dfl)					
Year	1.000–2.000	2.000–3.000	3.000–5.000	5.000–10.000	over 10.000
1675	22	7	5	4	5
1724	4	4	-	4	3
1750	12	3	2	2	-
1808	9	2	-	-	-

Table 13.4 Decline of wealth through the ages.

The meaning of the commands are explained table 13.5.

Command	Meaning
<code>\startxtable ... \stopxtable</code>	begin end table
<code>\startxrow ... \stopxrow</code>	begin end row
<code>\startxcell ... \stopxcell</code>	begin end column
<code>\startxtablehead ... \stopxtablehead</code>	begin end tablehead
<code>\startxtablebody ... \stopxtablebody</code>	begin end tablebody
<code>\startxtablefoot ... \stopxtablefoot</code>	begin end tablefoot
<code>\setupxtable</code>	table setup

Table 13.5 Commands to define extreme tables.

More information and examples can be found in the *Extreme Tables* manual.

13.4 Placing tables

In all examples you see the command `\placetable`. This command has the same function as `placefigure`. It takes care of the vertical spacing and numbering. The float mechanism is invoked and the table will end up on the most optimal location in your document.

`\placeFLOAT`

You can also set up the layout of tables with:

```
\setupfloats
```

You can set up the numbering and the labels with:

```
\setupcaptions
```

13

These commands are typed in the set up area of your input file and have a global effect on all floating blocks.

```
\setupfloats[location=left]
\setupcaptions[style=boldslanted,location={right,middle}]
\placetable[here][tab:opening hours]{Library opening hours.}
{\bTABLE[offset=4pt]
\bTR \bTD \bf Day \eTD \bTD[nx=2,align=middle] \bf Opening hours \eTD \eTR
\bTR \bTD Monday \eTD \bTD 14.00 -- 17.30 \eTD \bTD 18.30 -- 20.30 \eTD \eTR
\bTR \bTD Tuesday \eTD \bTD \eTD \bTD \eTD \eTR
\bTR \bTD Wednesday \eTD \bTD 10.00 -- 12.00 \eTD \bTD 14.00 -- 17.30 \eTD \eTR
\bTR \bTD Thursday \eTD \bTD 14.00 -- 17.30 \eTD \bTD 18.30 -- 20.30 \eTD \eTR
\bTR \bTD Friday \eTD \bTD 14.00 -- 17.30 \eTD \bTD \eTD \eTR
\bTR \bTD Saturday \eTD \bTD 10.00 -- 12.30 \eTD \bTD \eTD \eTR
\eTABLE}
```

The result is displayed in table 13.6.

Day	Opening hours	
Monday	14.00 - 17.30	18.30 - 20.30
Tuesday		
Wednesday	10.00 - 12.00	14.00 - 17.30
Thursday	14.00 - 17.30	18.30 - 20.30
Friday	14.00 - 17.30	
Saturday	10.00 - 12.30	

Table 13.6 *Library opening hours.*

14 | Tabulation / Paragraph formatting

Sometimes you want to typeset paragraphs in a specific formatted way. This is done with:

```
\startTABULATE
```

The tabulation mechanism is closely related to the table mechanism. You can use the tabulation mechanism in cases you want to typeset complete paragraphs within a cell. The tabulation mechanism also works fine at a page break.

A tabulate definition could look like this:

```
\starttabulate[|w(1.5cm)B|p(6.0cm)|p|]
\NC 1252
  \NC Hasselt obtains its city charter from bishop Hendrik
    van Vianden.
  \NC Hendrik van Vianden was pressed by other towns not
    to agree with the charter. It took Hasselt a long
    period of time to convince the Bishop. After
    supporting the Bishop in a small war against the
    Drents, the charter was released. \NC\NR
\NC 1350
  \NC Hasselt joins the Hanzepact to protect their
    international trade.
  \NC The Hanzepact was of great importance for merchants
    in Hasselt. In those days trading goods were taxed
    at every city, highway or rivercrossing. After
    joining the Hanzepact duty free routes all over
    Europe became available to Hasselt. However
    important the Hanzepact was, Hasselt always stayed a
    minor member of the pact. \NC\NR
\stoptabulate
```

In this case the first column is 1.5 cm wide and is typeset bold (B). The second column has a width of 6 cm and is typeset like a paragraph. The remaining horizontal space is used up by the last paragraph.

The example is typeset like this:

- 1252 Hasselt obtains its city charter from bishop Hendrik van Vianden. Hendrik van Vianden was pressed by other towns not to agree with the charter. It took Hasselt a long period of time to convince the Bishop. After supporting the Bishop in a small war against the Drents, the charter was released.
- 1350 Hasselt joins the Hanzepact to protect their international trade. The Hanzepact was of great importance for merchants in Hasselt. In those days trading goods were taxed at every city, highway or river-crossing. After joining the Hanzepact duty free routes all over Europe became available to Hasselt. However important the Hanzepact was, Hasselt always stayed a minor member of the pact.

14

The tabulation entries are placed between the `\start ... \stoptabulate` pair. Between the bracket pair you can specify the tabulate format with the column separators `|` and the format keys (see table 14.1).

Key	Meaning	Key	Meaning
<code>l</code>	left align	<code>I</code>	<i>italic</i>
<code>c</code>	center	<code>R</code>	<i>roman</i>
<code>r</code>	right align	<code>S</code>	<i>slanted</i>
<code>in</code>	spacing left	<code>T</code>	teletype
<code>jn</code>	spacing right	<code>m</code>	in-line math
<code>kn</code>	spacing around	<code>M</code>	display math
<code>w(d)</code>	1 line, fixed width	<code>f\command</code>	font specification
<code>p(d)</code>	paragraph, fixed width	<code>b{..}</code>	place .. before the entry
<code>p</code>	paragraph, maximum width	<code>a{..}</code>	place .. after the entry
<code>B</code>	boldface	<code>h\command</code>	apply \command on the entry

Table 14.1 Formatting keys for tabulate.

In table 14.2 you find an overview of the tabulate structuring commands.

Another example of paragraph formatting could look like this.

```
\definetabulate[ChemPar][|l|p|l|]
\startChemPar
\NC Limekilns
\NC Hasselt has its own limekilns. These were build in 1504
and produced quick lime up to 1956. Nowadays they are a
tourist attraction.
\NC \inlinechemical{CaCO_3,GIVES,CaO,+,CO_2} \NC\NR
\stopChemPar
```

Command		Meaning
<code>\start ... \stoptabulate</code>		begin end tabulate
<code>\NC</code>	next column	next column
<code>\NR</code>	next row	next row
<code>\HL</code>	horizontal line	horizontal line
<code>\TB</code>	table blank	empty line
<code>\definetabulate</code>		define own tabulate
<code>\setuptabulate</code>		tabulate setup

Table 14.2 Commands to define tabulate.

And it would come out like this:

Limekilns Hasselt has its own limekilns. These were build in 1504 $\text{CaCO}_3 \rightarrow \text{CaO} + \text{CO}_2$ and produced quick lime up to 1956. Nowadays they are a tourist attraction.

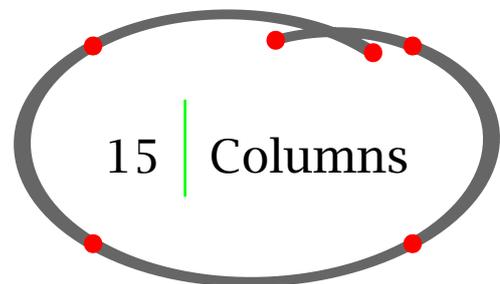
In chapter 9 your can find some more information on chemistry and `CONTEXT`.

Here we also introduced the command to define our own paragraph layout.

`\definetabulate`

and we also have:

`\setuptabulate`



Simple sections of text can be typeset in columns. If you precede a text fragment by `\startcolumns` and close the text fragment by `\stopcolumns` everything in between will be set in columns.

```
\startcolumns
```

Let's give an example:

```
\startcolumns[n=3,tolerance=verytolerant]
  Hasselt is an old Hanseatic City, situated 12~km north of Zwolle
  at the river Zwartewater.
```

```
...
```

```
Furthermore some events of special interest should be
mentioned. Every year at the end of August Hasselt celebrates
the \quote{Eui Festival} (hay festival).
```

```
\stopcolumns
```

15

The result will be a three column text.

Hasselt is an old Hanseatic City, situated 12 km north of Zwolle at the river Zwartewater.

The city has a long history since obtaining the city charter around 1252. Part and parcel of this history can be traced back to a large number of monuments to be admired in the city center.

There you will find the St. Stephanus church, a late gothic church dating back to 1479 with a magnificent organ. The former Municipal Building is situated on The Market Place. Constituted between 1500 and 1550 it houses a large collection of weapons, amongst which one of the largest collection of black powder guns (haakhussen) in the whole world should be mentioned.

Furthermore there is a corn windmill 'The Swallow', dating back to 1748 as well as the 'Stenendijk', a unique embankment and the last shell limekiln in Europe still in full

operation.

The city center with the townmoat adorned by lime-trees, the Van Stolkspark and the hustle and bustle at the docks are ideally suited for a stroll.

The area around Hasselt is also worth mentioning. In wintertime polder Mastebroek harbours large numbers of geese. In summertime the hamlets Genne, Streukel and Cellemuiden form, together with the very rare lapwing flowers (Lat. *Fritillaria meleagris*) found on the banks of the river Zwartewater, the ideal surroundings for walking or cycling trips.

Hasselt also is a very important center for watersports. The lakes of northwest Overijssel, the river IJssel, the Overijsselse Vecht and the Randmeren are within easy reach from the yacht harbour 'De Molenwaard'. Sailing, fishing, swimming and canoeing can be fully enjoyed in Hasselt.

Furthermore some events of special interest should be mentioned. Every year at the end of August Hasselt celebrates the 'Eui Festival' (hay festival).

If possible a new column can be enforced with `\column`. You can set up columns with:

```
\setupcolumns
```

In most cases you will obtain a better result by type setting the text on 'grid'. This is done by typing `grid=yes` in the command `\setuplayout`.

If you want to use columns within a framed text `\start ... \stopframedtext` there is the simple column mechanism.

```
\startframedtext[background=color,backgroundcolor=gray]
\startsimplecolumns
  In Hasselt's local newspaper there was a column on the
  local customs during New Years Eve.
  ...
  \midaligned{\inlinechemical{CaC_2,+2H_2O,GIVES,C_2H_2(g),+,Ca(OH)_2}}
  ...
  Nowadays the heavy metal lid of the milk can is replaced by
  a football. This does not reduce the sound but it is much
  saver.
\stopsimplecolumns
```

This will result in:

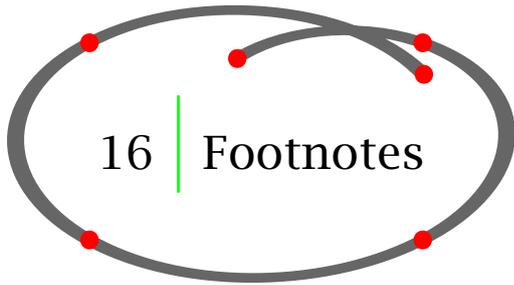
In Hasselt's local newspaper there was a column on the local customs during new years Eve. Next to the more general custom of eating Dutch doughnuts (oliebollen) and lighting fireworks there is the carbide shooting. What you need is an oldfashioned metal milk can, carbide, a little water and a lighter. The carbide and water is mixed in the closed milk can and will produce C_2H_2 gas (acetylene), via:



The volatile acetylene gas in the milk can is ignited via a small opening in the can. The result is a very loud detonation and the lid flies off.

It will not surprise you that Hasselts youth has a designated shooting ground for carbide shooting. Nowadays the heavy metal lid of the milk can is replaced by a football. This does not reduce the sound but it is much saver!

There is an advanced column mechanism available that is described in the *Columns* manual.



If you want to annotate your text you can use `\footnote`. The command looks like this:

`\footnote`

16

The bracket pair is optional and contains a logical name. The curly braces contain the text you want to display at the foot of the page.

The same footnote number can be called with its logical name.

`\note`

If you have typed this text:

The Hanse was a late medieval commercial alliance of towns in the regions of the North and the Baltic Sea. The association was formed for the furtherance and protection of the commerce of its members.`\footnote[war]{This was the source of jealousy and fear among other towns that caused a number of wars.}` In the Hanse period there was a lively trade in all sorts of articles such as wood, wool, metal, cloth, salt, wine and beer.`\note[war]` The prosperous trade caused an enormous growth of welfare in the Hanseatic towns.`\footnote{Hasselt is one of these towns.}`

It would look like this:

The Hanse was a late medieval commercial alliance of towns in the regions of the North and the Baltic Sea. The association was formed for the furtherance and protection of the commerce of its members.⁴ In the Hanse period there was a lively trade in all sorts of articles such as wood, wool, metal, cloth, salt, wine and beer.⁴ The prosperous trade caused an enormous growth of welfare in the Hanseatic towns.⁵

⁴ This was the source of jealousy and fear among other towns that caused a number of wars.

⁵ Hasselt is one of these towns.

Footnotes

The footnote numbering is done automatically. The command `\setupfootnotes` enables you to influence the display of footnotes:

```
\setupfootnotes
```

Footnotes can be set at the bottom of a page but also at other locations, like the end of a chapter. This is done with the command:

```
\placefootnotes
```

The footnotes will be placed at the end of your document with `\setupfootnotes[location=text]` in combination with `\placefootnotes` at the desired location.

You can also couple footnotes to a table. In that case we speak of local footnotes. The commands are:

```
\startlocalfootnotes
```

```
\placelocalfootnotes
```

An example illustrates the use of local footnotes:

```
\placetable[][productivity]
{Decline of Hasselt's productivity.\footnote{Source: {\em Uit
de geschiedenis van Hasselt.}}}
{\startlocalfootnotes
\starttable[|c|c|c|c|c|]
\HL
\NC
\NC Ovens
\NC Blacksmiths
\NC Breweries
\NC Tile works\footnote{The factories that produced roof tiles.} \NC\SR
\HL
\NC 1682 \NC 15 \NC 9 \NC 3 \NC 2 \NC\FR
\NC 1752 \NC ~6 \NC 4 \NC 0 \NC 0 \NC\LR
\HL
```

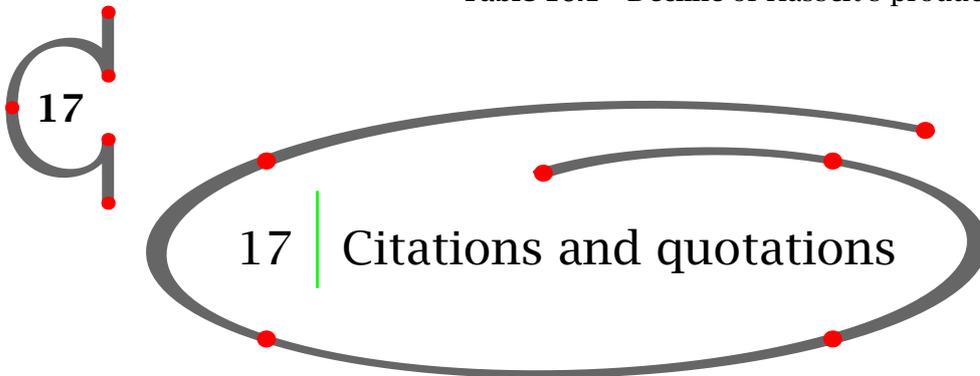
```
\NC \use5 \JustLeft{\placelocalfootnotes} \NC\FR
\stoptable
\stoplocalfootnotes}
```

This will result in table 16.1 with a local footnote. The footnote in the caption will appear at the bottom of the page.

	Ovens	Blacksmiths	Breweries	Tile works ¹
1682	15	9	3	2
1752	6	4	0	0

¹ The factories that produced roof tiles.

Table 16.1 Decline of Hasselt's productivity.⁶



The consistent use of quote and quotation marks in the running text is invoked by the use of `\quote` or `\quotation`. For longer text fragments you can use:

```
\startquotation
```

In the book `\quote{Hasselt, beelden van een middeleeuwse stad}` it says:

```
\startquotation
```

Het stadhuis wordt voor het eerst vermeld in 1431. Oorspronkelijk is het een houten huis, dat wordt afgebroken om plaats te maken voor een nieuw stadhuis van steen. Dit wordt echter halverwege de 16e eeuw ook afgebroken en vervangen door een nog groter pand. Het nieuwe stadhuis wordt weer in dezelfde fraaie stijl opgebouwd. De bestuurders laten daarmee zien dat het is gebouwd in een tijd van grote welvaart.

```
\stopquotation
```

In the example below you can see that quotation is language sensitive:

⁶ Source: *Uit de geschiedenis van Hasselt*.

Definitions

```
\nl Hij zei tegen me: \quotation{In Hasselt noemen ze dat
\quote{noaberschap} of zoiets.}
\n He told me: \quotation{In Hasselt they call this
\quote{noaberschap} or something like that.}
\de Er sagte zu mir: \quotation{In Hasselt nennt man das
\quote{noaberschap} oder so etwas.}
\fr Il a dit: \quotation{À Hasselt on c'appelle \quote{noaberschap}
ou quelque chose comme ça.}
```

Note the automatic change of the quotation marks in case of a quote within a quote.

Hij zei tegen me: „In Hasselt noemen ze dat ‚noaberschap‘ of zoiets.”

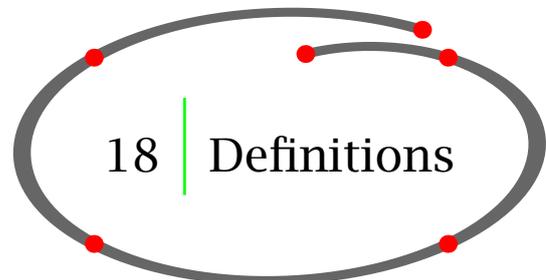
He told me: “In Hasselt they call this ‘noaberschap’ or something like that.”

Er sagte zu mir: „In Hasselt nennt man das ‚noaberschap‘ oder so etwas.“

Il a dit: « À Hasselt on c'appelle «noaberschap» ou quelque chose comme ça. »

You can alter the default settings with:

```
\setuplanguage
```



If you want to display notions, concepts and ideas in a consistent manner you can use:

```
\definedescription
```

For example:

```
\definedescription
[concept]
[alternative=serried,headstyle=bold,width=broad]
\concept{Hasselter juffer} A sort of biscuit made of puff pastry and
covered with sugar. It tastes very sweet. \par
```

It would look like this:

Hasselter juffer A sort of biscuit made of puff pastry and covered with sugar. It tastes very sweet.

But you can also choose other layouts:

Hasselter bitter

A very strong alcoholic drink (up to 40%) mixed with herbs to give it a special taste. It is sold in a stone flask and it should be served ijskoud (as cold as ice).

Euifeest

A harvest home to celebrate the end of a period of hard work. The festivities take place in the last week of August.

If you want to avoid the `\par` or when you have more than one paragraph in the definition you can use the `\start... \stop` construct.



```
\definedescription
  [concept]
  [alternative=right,
   headstyle=bold,
   width=broad]
```

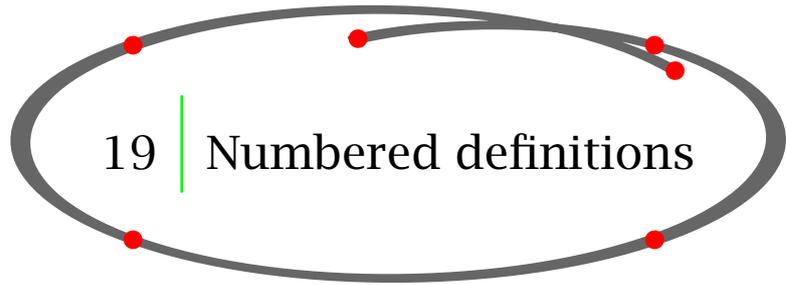
```
\startconcept{Euifeest} A harvest home to celebrate the end of a
period of hard work.
This event takes place at the end of August and lasts one week. The
city is completely illuminated and the streets are decorated. This
feast week ends with a {\em Braderie}.
\stopconcept
```

This would become:

A harvest home to celebrate the end of a period of hard work. This event takes **Euifeest** place at the end of August and lasts one week. The city is completely illuminated and the streets are decorated. This feast week ends with a *Braderie*.

Layout is set up within the second bracket pair of `\definedescription[] []`. But you can also use:

```
\setupdescriptions
```



With `\defineenumeration` you can number text elements like remarks or questions. If you want to make numbered remarks in your document you use:

```
\defineenumeration
```



For example:

```
\defineenumeration
  [remark]
  [alternative=top,
   text=Remark,
   inbetween={\blank[none]},
   after=\blank]
```

Now the new commands `\remark`, `\subremark`, `\resetremark` and `\nextremark` are available and you can type remarks like this:

```
\remark In the early medieval times Hasselt was a place of
pilgrimage. The {\em Heilige Stede} (Holy Place) was torn down during
the Reformation. In 1930, after 300 years, the {\em Heilige Stede} was
reopened.
```

```
\subremark Nowadays the {\em Heilige Stede} is closed again but once
a year an open air service is held on the same spot. \par
```

This becomes:

Remark 1

In the early medieval times Hasselt was a place of pilgrimage. The *Heilige Stede* (Holy Place) was torn down during the Reformation. In 1930, after 300 years, the *Heilige Stede* was reopened.

Remark 1.1

Nowadays the *Heilige Stede* is closed again but once a year an open air service is held on the same spot.

You can reset numbering with `\resetremark` or `\resetsubremark` or increment a number with `\nextremark` or `\nextsubremark`. This is normally done automatically per chapter, section or whatever.

Outlined text

You can set up the layout of `\defineenumeration` with:

```
\setupenumerations
```

You can also vary the layout of remark and subremark in the example above with:

```
\setupenumeration[remark][headstyle=bold]
\setupenumeration[subremark][headstyle=slanted]
```

If a number becomes obsolete you can type:

```
\remark[-]
```

If the remark contains several paragraphs you should use the command pair `\start ... \stopremark` that becomes available after defining remark with `\defineenumeration[remark]`.

20

```
\setupenumeration
  [remark]
  [alternative=hanging,
   width=broad]
```

```
\startremark
```

In the early medieval times Hasselt was a place of pilgrimage. The `{\em Heilige Stede}` (Holy Place) was torn down during the Reformation.

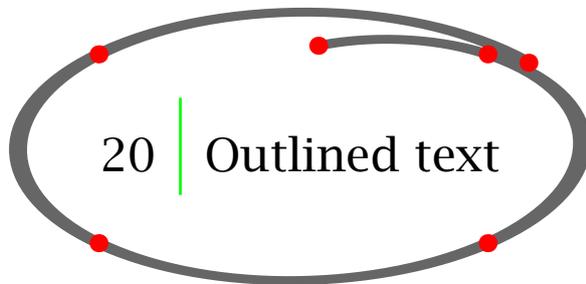
After 300 years in 1930 the `{\em Heilige Stede}` was reopened. Nowadays the `{\em Heilige Stede}` is closed again but once a year an open air service is held on the same spot.

```
\stopremark
```

So the example above would look like this:

Remark 2 In the early medieval times Hasselt was a place of pilgrimage. The *Heilige Stede* (Holy Place) was torn down during the Reformation.

After 300 years in 1930 the *Heilige Stede* was reopened. Nowadays the *Heilige Stede* is closed again but once a year an open air service is held on the same spot.

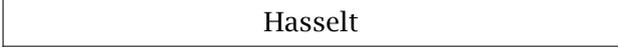


You can `outline` a text with `\framed`. The command looks like this:

`\framed`

The bracket pair is optional and contains the set up parameters. The curly braces enclose the text. To be honest, the outlined text in the first paragraph was done with `\inframed`. This command takes care of the interline spacing.

Some other examples of `\framed` and its set up parameters are shown below.

<code>\framed</code> [height=fit, width=.5\textwidth] {Hasselt}	
<code>\framed</code> [height=3em, width=.5\textwidth] {Hasselt now has more space}	
<code>\framed</code> [height=3em, width=.5\textwidth, foregroundcolor=red, framecolor=blue] {Hasselt now has some color}	
<code>\framed</code> [height=3em, width=.5\textwidth, foregroundcolor=red, framecolor=blue, rulethickness=2pt] {Hasselt now has more frame}	
<code>\framed</code> [height=3em, width=.5\textwidth, foregroundcolor=red, framecolor=blue, rulethickness=2pt, background=color, backgroundcolor=green] {Hasselt now has a colorful background}	
<code>\framed</code> [height=3em, width=.5\textwidth, foregroundcolor=red, framecolor=blue,	

Outlined paragraphs

```
rulethickness=2pt,  
background=color,  
backgroundcolor=green,  
foregroundstyle=bold]  
{Hasselt now has another style}  
\framed  
[height=3em,  
width=.5\textwidth,  
foregroundcolor=red,  
framecolor=blue,  
rulethickness=2pt,  
background=linear shade,  
foregroundstyle=bold]  
{Hasselt now has a little shade}
```



Hasselt now has a little shade

21

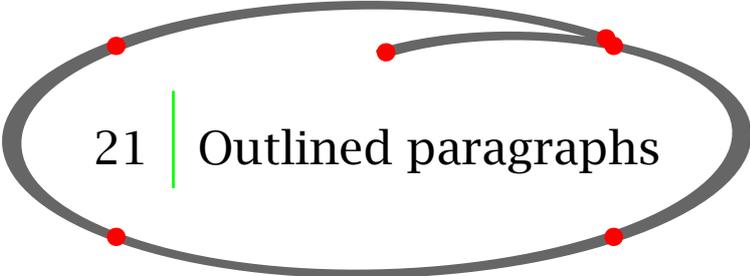
The shady background was defined with:

```
\definecolor[a][black]  
\definecolor[b][white]  
\startuniqueMPgraphic{LinearShade}  
fill OverlayBox  
withshademethod "linear" withcolor \MPcolor{a} shadedinto \MPcolor{b}  
;  
\stopuniqueMPgraphic  
\defineoverlay  
[linear shade]  
[\uniqueMPgraphic{LinearShade}]
```

The `\framed` command is very sophisticated and is used in many macros. The command to set up frames is:



`\setupframed`



21 | Outlined paragraphs

Complete paragraphs can be outlined with:

```
\startFRAMEDTEXT
```

Let's give an example:

```
\definefloat[intermezzo]
\setupframedtexts
  [width=.8\makeupwidth,
   background=color,
   backgroundcolor=gray,
   corner=round,
   framecolor=blue,
   rulethickness=2pt]

\placeintermezzo[here][block:bridge]{An intermezzo.}
\startframedtext
  It was essential for Hasselt to have a bridge across the Zwarte
  Water river. The bishop of Utrecht gave Hasselt his consent in
  1486.
  \blank
  Other cities in the neighbourhood of Hasselt were afraid of the
  toll money to be paid when crossing this bridge so they
  prevented the construction for many years.
\stopframedtext
```



This example also illustrates the command `\definefloat`. You can find more information on this command in paragraph 40.5. The `\blank` is necessary to enforce a blank line.

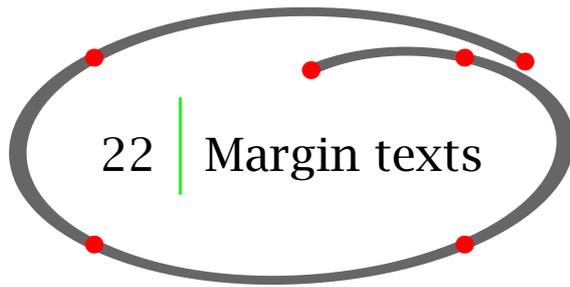
It was essential for Hasselt to have a bridge across the Zwarte Water river. The bishop of Utrecht gave Hasselt his consent in 1486.

Other cities in the neighbourhood of Hasselt were afraid of the toll money to be paid when crossing this bridge so they prevented the construction for many years.

Intermezzo 21.1 An intermezzo.

The outlining can be set up with:

```
\setupframedtexts
```



It is very easy to put text in the margin. You just use `\inmargin`.

`\inmargin`

You may remember one of the earlier examples:

```
\inmargin
  {\externalfigure
   [ma-cb-23]
   [width=.6\marginwidth]}
```

This would result in a figure in the margin. You can imagine that it looks quite nice in some documents. But be careful. The margin is rather small so the figure could become very marginal.

A few other examples are shown in the text below.

The Ridderstraat (Street of knights) `\inmargin{Street of\\Knights}` is an obvious name. In the 14th and 15th centuries, nobility and prominent citizens lived in this street. Some of their big houses were later turned into poorhouses `\inright{poorhouse}` and old peoples homes.

Up until `\inleft[low]{\tfc 1940}`1940 there was a synagog in the Ridderstraat. Some 40 Jews gathered there to celebrate their sabbath. During the war all Jews were deported to Westerbork and then to the extermination camps in Germany and Poland. None of the Jewish families returned. The synagog was knocked down in 1958.

The commands `\inmargin`, `\inleft` and `\inright` all have the same function. In a two sided document `\inmargin` puts the margin text in the correct margin. The `\\` is used for line breaking. The example above would look like this:

**Street of
Knights**

The Ridderstraat (Street of knights) is an obvious name. In the 14th and 15th centuries, nobility and prominent citizens lived in this street. Some of their big houses were later turned into poorhouses and old peoples homes.

poorhouses

1940

Up until 1940 there was a synagog in the Ridderstraat. Some 40 Jews gathered there to celebrate their sabbath. During the war all Jews were deported to Westerbork and then to the extermi-

nation camps in Germany and Poland. None of the Jewish families returned. The synagog was knocked down in 1958.

You can set up the margin text with:

```
\setupmargin
```

Other commands that you can use for forcing text into the margin are listed in table 22.1.

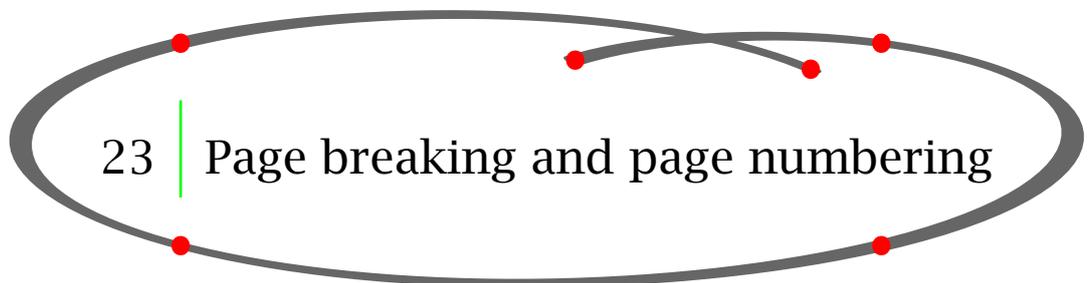
Command	Meaning
<code>\ininner</code>	text in inner margin
<code>\inouter</code>	text in outer margin
<code>\inright</code>	text in right margin
<code>\inleft</code>	text in left margin
<code>\inmargin</code>	text in the margin
<code>\inothmargin</code>	text in other margin
<code>\margin</code>	text in the margin



Table 22.1 Overview of margin commands.

If you want to place more extensive text blocks in the margin there is the command: and the accompanying command:

```
\setupmarginblocks
```



23.1 Page break

A page can be enforced or blocked by:





`\page`

The options can be stated within the brackets. The options and their meaning are presented in table 23.1.

Option	Meaning
yes	enforce a page
makeup	enforce a page without filling
no	no page
preference	prefer a new page here
bigpreference	great preference for a new page here
left	next page is a left handside page
right	next page is a right handside page
disable	following commands have no effect
last	add pages till even number is reached
quadruple	add pages till a multiple of four is reached
even	next page is even
odd	next page in odd
blank	no page number
empty	insert an empty page
reset	following commands do have effect
start	from now on page commands have effect
stop	from now on page commands have no effect

Table 23.1 Page options.

23.2 Page numbering

Numbering pages is done automatically by `CONTEXT`. However, numbering the pages the way you want it may take some effort.

A rather simple `\start ... \stoptext` document will be numbered from $1..n$ (where n is the last page). If you want your document to number its pages alphabetical you can type:

```
\setupuserpagenumber
  [numberconversion=character]
```

in the setup area of your file.

You can enforce a page number with:

```
\setupuserpagenumber [number=25]
```

The options of the `\setupuserpagenumber` command are given in table 23.2.

Option	Meaning
way	how to number the document
prefix	use pagenumber prefix
prefixset	use defined prefixset
prefixseparatorset	use defined separator
state	start - stop page numbering
number	define page number
numberconversion	convert page number
numberconversionset	used defined conversion set

Table 23.2 Page numbering: numbering options.

The `prefixset`, `prefixseparatorset` and the `numberconversionset` options are defined with the `\defineprefixset`, `\defineseparatorset` and `\defineconversionset` respectively.

This manual uses the `CONTEXT` standard document section blocks: `frontpart`, `bodymatter` and `appendices`. These section blocks are numbered with roman characters, numeral digits and characters respectively.

```
\defineconversionset
  [frontpart:pagenumber] [] [romannumerals]
```

```
\defineconversionset
  [bodypart:pagenumber] [] [numbers]
```

```
\defineconversionset
  [appendix:pagenumber] [] [Characters]
```

At the start of each section block the number is reset to `i`, `1` and `A` respectively.

The same effect would have been obtained with:

```
\startsectionblockenvironment[frontpart]
  \setupuserpagenumber[numberconversion=romannumerals]
\stopsectionblockenvironment
```

Page numbering and the location of the page numbers can be set up with:

```
\setuppagenumbering
```

The options of this command are shown in table 23.3:

Note that this is also the command that indicates that your document is single or double sided which has an effect on the left-right page layout.

```
\setuppagenumbering
  [alternative=doublesided]
```

Option	Meaning
alternative	page layout: single or double sided
location	location of page number on page
width	width of page number
left	text left of page number
right	text right of page number
page	...
state	start - stop page numbering
command	invoke command
style	set character style
color	set color

Table 23.3 Page numbering: layout options.

24

In this manual page numbering is set up with:

```
\setuppagenumbering
 [location={footer,middle},
  command=\NummerCommando]
```

The `\NummerCommando` uses METAPOST to draw a unique random image around each page number.

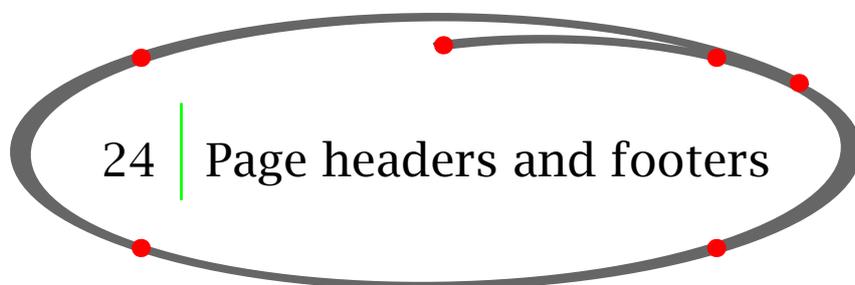
You can recall a page number with `\userpagenumber`. If you set up your headertext with:

```
\setupheadertexts
 [Page \userpagenumber\ of \lastuserpagenumber]
```

You would get a header with the actual page number and the total of pages (in that section block).

The actual page number and the real page number may differ since there may be pages or sections that in your document that are not numbered. If you feel the need to display the real page number there is the command `\realpagenumber`.

Please refer to the [CONTEXT WIKI](#) for more details.



In some cases you want to give your document a page header and footer. The commands to do this are:

Table of contents (lists)

```
\setupfootertexts
```

```
\setupheadertexts
```

The first bracket pair is used for the location of the footer or header (`text`, `edge` etc). Footer and header are placed within the second and third bracket pairs. In a double sided document a fourth and fifth bracket pair is used for footer and header on the left-hand side page and the right-hand side page. In most cases you can omit these last two bracket pairs.

```
\setupfootertexts[Manual][section]
```

In this case the text *Manual* will appear in the left-hand side corner and the title of the actual section on the right-hand side of the page. This footer will change with the beginning of a new section.

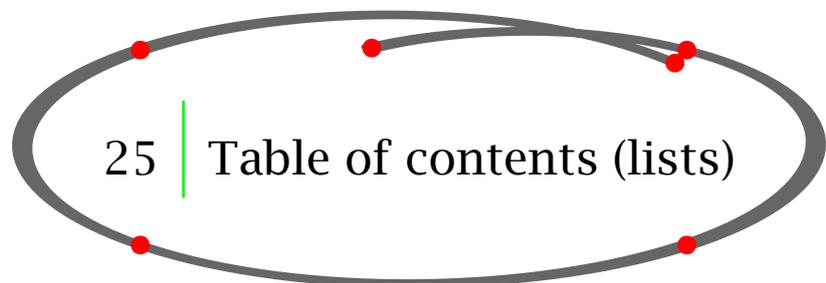
You can set up the layout of the header and footer with:

```
\setupheader
```

```
\setupfooter
```

If you want to leave out the page header and footer you can type:

```
\noheaderandfooterlines
```



A table of contents contains chapter numbers, chapter titles and page numbers and can be extended with sections, sub sections, etc. A table of contents is generated automatically by typing:

`\placecontent`

Which table of contents is produced depends on the location of this command in your document. At the start of the document it will generate a list of chapters, sections etc. But at the top of a chapter:

```
\chapter{Hasselt in Summer}
\placecontent
\section{Hasselt in July}
\section{Hasselt in August}
```

it will only produce a list of (sub) section titles with the corresponding section numbers and page numbers.

The predefined command `\placecontent` is available because it was defined with:

`\definecombinedlist`

25

This command and `\definelist` allows you to define your own lists necessary for accessing your documents.

The use of this command and its related commands is illustrated for the default available table of contents.

```
\definelist[chapter]
\setuplist
  [chapter]
  [before=\blank,
   after=\blank,
   style=bold]
\definelist[section]
\setuplist
  [section]
  [alternative=d]
```

Now there are two lists of chapters and sections and these will be combined in a table of contents with the command `\definecombinedlist`.

```
\definecombinedlist
  [content]
  [chapter,section]
  [level=subsection]
```

Now two commands are available: `\placecontent` and `\completecontent`. With the second command the title of the table of contents will be added to the table of contents.

The layout of lists can be varied with the parameter `alternative`.

Table of contents (lists)

Alternative	Display
a	number - title - page number
b	number - title - spaces - page number
c	number - title - dots - page number
d	number - title - page number (continuing)
e	reserved for interactive purposes
f	reserved for interactive purposes
g	reserved for interactive purposes

Table 25.1 Alternatives for displaying lists.

Lists are set up with:

```
\setuplist
```

```
\setupcombinedlist
```

25

If you want to change the layout of the generated table of contents you'll have to remember that it is a (combined) list and that we can set the partial lists separately.

```
\setuplist  
  [section]  
  [textstyle=bold,  
  pagestyle=bold,  
  numberstyle=bold]
```

This will result in a bold page number, section title and section number.

Lists are generated and placed with:

```
\placelist
```

So if you want a list of sections at the beginning of a new chapter, you type:

```
\placelist[section]
```

only the sections will be displayed.

A long list or a long table of contents will use up more than one page. To be able to force page breaking you can type:

```
\placecontent[extras={8.2=page}]
```

A page break will then occur after section 8.2.

In some cases you want to be able to write your own text in an automatically generated list. This is done with:

```
\writetolist
```

```
\writebetweenlist
```

For example if you want to make a remark in your table of contents after a section titled *Hotels in Hasselt* you can type:

26

```
\section{Hotels in Hasselt}
\writebetweenlist[section]{\blank}
\writetolist[section][location=here]{{Section under construction}}
\writebetweenlist[section]{\blank}
```



It is possible to generate one or more registers. By default the command `\index` is available. If you want to add a word to the index you type:

```
\index{town hall}
```

The word *town hall* will appear as an index entry in the sorted register. Sometimes the index word does not appear in normal alphabetic order. For example, entries such as symbols have to provide extra sorting information in order to produce a correct alphabetical list:

```
\index[minus]{$-$}
```

Sometimes you have sub- or sub sub entries. These can be defined as follows:

```
\index{town hall+location}
```

```
\index{town hall+architecture}
```

You can generate your register with the command:

Synonyms

`\placeindex`

or

`\completeindex`

The command `\index` is a predefined `CONTEXT` command, but of course you can also define your own registers.

`\defineregister`

For example if you want to make a new register based on the streets in Hasselt you could type:

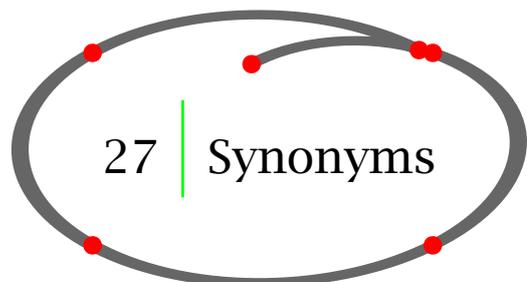
`\defineregister[street]`

Now a new register command `\street` is available. Now `\street{Ridderstraat}` is a new index entry. To produce a list of entries you could now use:

`\placeregister[street]`
`\placestreet`
`\completetestreet`

You can alter the layout of the registers with:

`\setupregister`



In many documents people want to use specific words consistently throughout the document. To enforce consistency the command below is available.

`\definesynonyms`

The first bracket pair contains the singular form of the synonym, and the second contains the plural form. The third bracket pair contains a command.



For example the command `\abbreviation` is defined by:

```
\definesynonyms[abbreviation][abbreviations][\infull]
\setupsynonyms[style=cap]
```

Now the command `\abbreviation` is available and can be used to state your abbreviations:

```
\abbreviation{ANWB}{Dutch Automobile Association}
\abbreviation{VVV}{Bureau of Tourist Information}
\abbreviation{NS}{Dutch Railways}
```

If you would type:

The Dutch `\VVV` (`\infull{VVV}`) can provide you with the tourist information on Hasselt.

You would obtain something like this:

The Dutch VVV (BUREAU OF TOURIST INFORMATION) can provide you with the tourist information on Hasselt.

The list of synonyms or abbreviations is best defined in the set up area of your input file for maintenance purposes. You can also store this kind of information in an external file, and load the file (e.g. `abbrev.tex`) with:

```
\input abbrev.tex
```

If you want to put a list of the abbreviations used in your document you can type:

```
\placelistofabbreviations
```

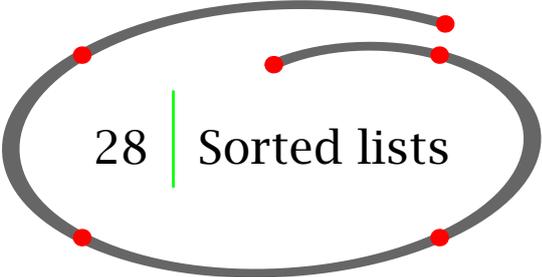
or

```
\completelistofabbreviations
```

A complete and sorted list with used abbreviations and their meaning is produced.

The typesetting of synonyms can be influenced with:

```
\setupsynonyms
```



28 | Sorted lists

If you want to create a sorted list you can use:

```
\definesorting
```

For example:

```
\define[1]\street{#1\Street{#1}}
\definesorting[Street][Streets]
\setupsorting[Street][criterium=all]
```

When you walk in the `\street{Eikenlaan}` you will cross the `\street{Vechtlaan}` and `\street{Gasthuisstraat}`. Go left into the `\street{Gasthuisstraat}` and take another left on the `\street{Heerengracht}`. You walk along the canal to the `\street{Ridderstraat}`, there you turn right. Cross the canal and turn left to the `\street{Julianakade}`. There you can enjoy the view over the Zwartewater.

So the streets you visited are:

```
\placelistofStreets
```

This will become:

When you walk in the Eikenlaan you will cross the Vechtlaan and Gasthuisstraat. Go left into the Gasthuisstraat and take another left on the Heerengracht. You walk along the canal to the Ridderstraat, there you turn right. Cross the canal and turn left to the Julianakade. There you can enjoy the view over the Zwartewater.

So the streets you visited are:

```
Eikenlaan
Gasthuisstraat
Heerengracht
Julianakade
Ridderstraat
Vechtlaan
```

Note that the Gasthuisstraat appears only once in the list.

The predefined `\logo` command is used for the consistent use of text logos.

When you define:

```
\logo [HSTEX] {Hassel\TeX}
```

You can use that logo throughout your text.

How would you call a `\TeX` based macropackage when you work in Hasselt? `\HSTEX`?

How would you call a `TEX` based macropackage when you work in Hasselt? `HASSELTEX`?

29 | Referring to text elements

To disclose your document for your readers you can use the table of contents and the register. However, it is not uncommon to refer to specific text elements like formulas, tables, images and sections to enhance readability.

For referring from one location in a document to another you can use the command:

```
\in
```

The curly braces contain text and the brackets contain a logical label. If you have written a chapter header like this:

29

```
\startchapter[title=Hotels in Hasselt,reference=hotel]
...
\stopchapter
```

then you can refer to this chapter with:

```
\in{chapter}[hotel]
```

After processing the chapter number is available and the reference could look something like: *chapter 23*. You can use `\in` for any references to text elements like chapters, sections, figures, tables, formulas etc.

Another example:

There are a number of things you can do in Hasselt:

```
\startitemize[n,packed]
\item swimming
\item sailing
\item[hiking] hiking
\item biking
\stopitemize
```

An activity like `\in{activity}[hiking]` described on `\at{page}[hiking]` is very tiring.

This would look like this:

There are a number of things you can do in Hasselt:

1. swimming
2. sailing

3. hiking
4. biking

An activity like activity 3 described on page 69 is very tiring.

As you can see, it is also possible to refer to pages. This is done with:

```
\at
```

For example with:

```
\at{page}[hiking]
```

This command can be used in combination with:

```
\pagereference
```

and

```
\textreference
```

If you want to refer to the chapter *Hotels in Hasselt* you could type:

```
Look in \in{chapter}[hotel] on \at{page}[hotel] for a complete
overview of accomodations in \pagereference[accomodation]Hasselt.
```

A chapter number and a page number will be generated when processing the input file. On another spot in the document you can refer to accomodation with `\at{page}[accomodation]`.

You can also define a set of labels separated by commas.

```
\placefigure
[here]
[fig:canals,fig:boats]
{A characteristic picture of Hasselt.}
{\externalfigure[ma-cb-08][width=10cm]}
```

There are many canals in Hasselt (see `\in{figure}[fig:canals]`).

.
. .
.

Boats can be moored in the canals of Hasselt (see `\in{figure}[fig:boats]`).

This might look like this:



Figure 29.1 A characteristic picture of Hasselt.

There are many canals in Hasselt (see figure 29.1). . . . Boats can be moored in the canals of Hasselt (see figure 29.1).

You can also refer to a title of a chapter or section or even a caption of an image. This is done with:

29

`\about`

This:

The caption of `\in{figure}[fig:canals]` is `{\em \about[fig:canals]}`.

Becomes:

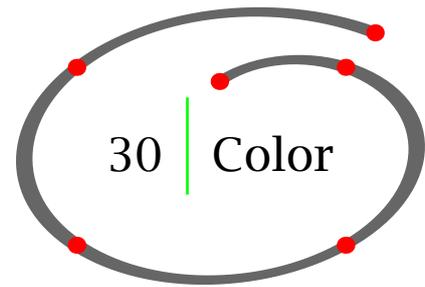
The caption of figure 29.1 is “*A characteristic picture of Hasselt.*”.

With the command:

`\setupinteraction[state=start]`

all references become active links. See chapter 32 for more information on this subject.

Color



Text, frames or backgrounds can be set in color with:

```
\color
```

Default the basic colors are available. Basic colors are for example red, white and blue. A color like orange can be defined with:

```
\definecolor
```

You can define orange like this:

```
\definecolor [darkorange] [c=0.0,m=0.60,y=1.00,k=0.0]  
\definecolor [middleorange] [.5(darkorange)]
```

It is of good practice to check (combinations of) colors on a larger surface:

```
\blackrule[width=\hsize,height=1cm,color=red,after=]  
\blackrule[width=\hsize,height=1cm,color=white,after=]  
\blackrule[width=\hsize,height=1cm,color=blue,after=]  
\blackrule[width=\hsize,height=1cm,color=darkorange]
```

so you can see if they fit together:



A color can be invoked in a number of ways:

```
\startcolor[red]
```

On `{\darkorange Kingsday}` `{\blue Hasselt}` turns into a

For Hasselt the 15th and 16th century were relatively unstable times. There were uprisings and disputes with neighbouring cities. To be able to defend themselves the city council ordered a number of arquebuses (very primitive firearms). Fourteen of these have survived and now form one of the greatest arquebus collections in Europe.

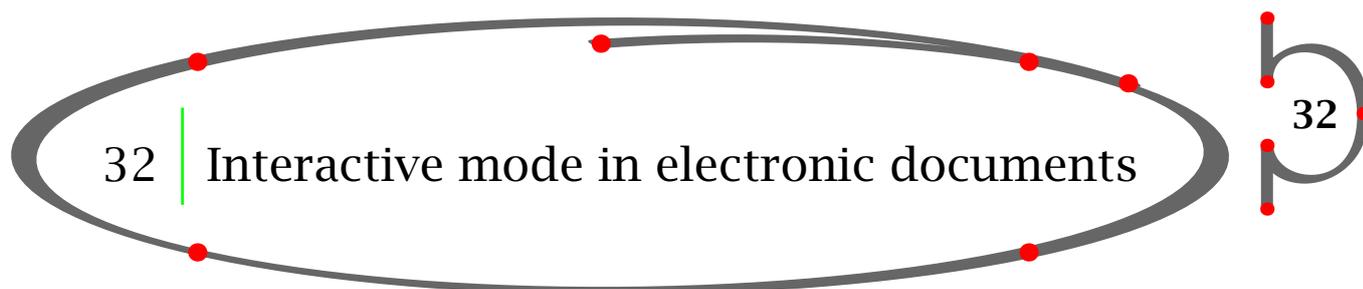
`\stopalignment`

This will become a rightaligned paragraph without hyphenations:

For Hasselt the 15th and 16th century were relatively unstable times. There were uprisings and disputes with neighbouring cities. To be able to defend themselves the city council ordered a number of arquebuses (very primitive firearms). Fourteen of these have survived and now form one of the greatest arquebus collections in Europe.

In case of alignment you can specify a tolerance and the direction (vertical or horizontal). Normally the tolerance is `verystrict`. In columns you could specify `verytolerant`. The tolerance in this manual is:

`\setuptolerance[horizontal,verystrict]`



32.1 Introduction

Documents that are electronically available for consulting and displaying on a computer screen are called interactive documents.

Interaction means that you can click on active areas and jump to the indicated locations. For example if you consult a register you can click on a (active) page number and you will jump to the corresponding page.

Interaction relates to:

- active chapter numbers in the table of content
- active page numbers in registers
- active page numbers, chapter numbers and figure numbers in internal references to pages, chapters, figures etc. in the running text
- active titles, page numbers, and chapter numbers in external references to other interactive documents
- active menus as navigation tools
- references to webpages and programs

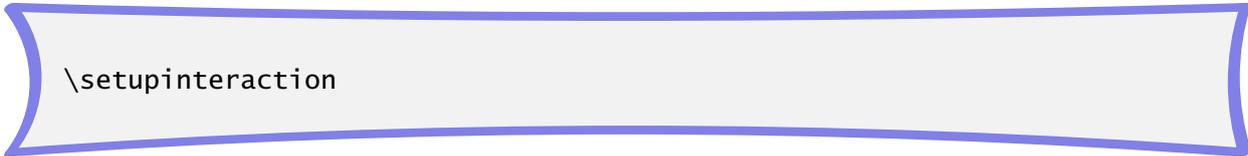
Interactivity depends on the program you use to view the interactive document. We assume here that you will use ACROBAT READER for viewing.

CONTEX_T is a very powerful system for producing electronic or interactive PDF documents. However, only a few standard features are described in this chapter. As the authors of this manual are planning to make all CONTEX_T related manuals electronically (sources included) available, reverse engineering is one of the options to become more acquainted with the possibilities of CONTEX_T.

Good examples of interactive documents are CONTEX_T presentations (see chapter 42). For more complex interactive PDF documents with forms you should read the Widgets manual.

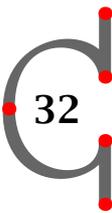
32.2 Interactive mode

The interactive mode is activated by:



```
\setupinteraction
```

For example:



```
\setupinteraction
  [state=start,
   color=green,
   style=bold]
```

The hyper links are now generated automatically and the active words are displayed in bold green.

The interactive document is considerably bigger (in MB's) than its paper cousin because hyperlinks consume space. You will also notice that processing time becomes longer. Therefore it is advisable to de-activate the interactive mode as long as your document is under construction.

32.3 Interaction within a document

Earlier you have seen how to make a reference with `\in` and `\at`. You may have wondered why you had to type `\in{chapter}[chap:introduction]`. In the first place *chapter* and its corresponding chapter number will not be separated at line breaking. In the second place the word *chapter* and its number are typeset differently in the interactive mode. This gives the user a larger clickable area.

32.4 Interaction between documents

It is possible to link one document to another. First you have to state that you want to refer to another document. This is done by:

```
\useexternaldocument
```

The first bracket pair must contain a logical name of the document, the second pair the file name of the other document and the third pair is used for the title of the document.

For referring to these other documents you can use:

```
\from
```

The curly braces contain text and the brackets contain the reference.

Look at the example below.

```
\useexternaldocument
[hia][hasselbook][Festivities in Hasselt]
Most tourist attractions are described in \from[hia].\crlf
A description of the \about[hia::euifeest] is found in \from[hia].\crlf
The euifeest is described on \at{page}[hia::euifeest] in \from[hia].\crlf
See for more information \in{chapter}[hia::euifeest] in \from[hia].
```

The `\useexternaldocument` is usually typed in the set up area of your input file.

After processing your input file and the file `hasselbook.tex`, you will have two PDF documents. The references come out like this:

Most tourist attractions are described in **Festivities in Hasselt**.
A description of the “” is found in **Festivities in Hasselt**.
The euifeest is described on **page** in **Festivities in Hasselt**.
See for more information **chapter** in **Festivities in Hasselt**.

For more information on cross referencing look at `CONTEXT Magazine 1103`.

32.5 Interaction with the world wide web

In interactive mode there is one other command that has little meaning in the paper version.

```
\goto
```

The curly braces contain text, the brackets contain a reference (logical name or a location).

```
In \goto {Hasselt} [ url(http://www.stadindex.nl/plattegrond/hasselt) ]
all streets are build in a circular way.
```

In the interactive document Hasselt will be green and active. When you click the text you will jump to a map of Hasselt.

For a consistent definition of the urls there is the command:

```
\useURL
```

The address is defined with:

```
\useURL  
  [loc:cityplan]                % id  
  [http://www.stadindex.nl/plattegrond/hasselt] % adress  
  []                             % document  
  []                             % text
```

The webaddress is recalled by its logical name: `\goto{Hasselt} [url(loc:cityplan)]`.

It is of good practice to define and maintain the urls in a separate file.

32.6 Buttons

The command to define a button is:

```
\button
```

The first bracket pair contains the setup keys, the curly brackets contain the button text and the last bracket pair the destination.

```
\useexternal soundtrack  
  [stranger][wayfaring_stranger.mp3]  
  
\button{Website Hasselt} [ url(http://www.hasselt.nl) ]  
\button{MSWord Document} [ program(hasselt.doc) ]  
\button{Sound Clip}      [ StartSound{stranger} ]
```

The first example results in a jump to a webpage, the second opens the file `hasselt.doc` in MS WORD and the third plays a tune. Note the use of the `\useexternal soundtrack` command.

32.7 Menus

You can define a menu with:

```
\startinteractionmenu
```

And set it up with:

```
\setupinteractionmenu
```

The first bracket pair is used for its name and the second pair for setting up the menu.

A menu can be used in an interactive document. Below you can find a simple example that you can copy to do some experimenting:

```
\setuppapersize
  [S6][S6]

\setuplayout
  [header=0cm,          topspace=.5cm, backspace=2cm,
   margindistance=.5cm, margin=1cm,    rightmargin=0cm,
   edgedistance=.5cm,  rightedge=2cm, width=fit,
   height=13.8cm,     footer=1cm,     bottom=1cm]

\setupinteraction
  [state=start,       menu=on]

\setupinteractionmenu
  [bottom]
  [background=color,  backgroundcolor=gray, frame=off]

\startinteractionmenu[bottom]
\hfill
\startbut [content]      contents      \stopbut \quad
\startbut [index]       index        \stopbut \quad
\startbut [PreviousJump] last location \stopbut \quad
\startbut [NextPage]    next page    \stopbut \quad
\startbut [CloseDocument] exit          \stopbut \quad
\stopinteractionmenu

\starttext

\startstandardmakeup
  \midaligned{\tfd Festivities in Hasselt}
\stopstandardmakeup

\completecontent

\startchapter[title=Introduction]
  An introduction.
\stopchapter

\startchapter[title=Kingsday]
  Something about Kingsday in Hasselt.\index{Kingsday}
\stopchapter
```

```

\startchapter[title=Hassailt]
  Something about Hassailt.\index{Hassailt}
\stopchapter

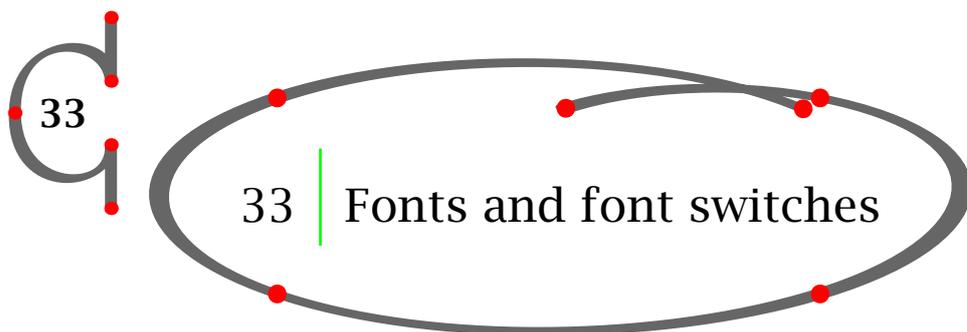
\startchapter[title=Euifeest,reference=euifeest]
  Something about the Euifeest.\index{Euifeest}
\stopchapter

\completeindex

\stoptext
    
```

The definition of the `\startinteractionmenu` will produce a menu at the bottom of every screen. The menu buttons contain the text *contents*, *index*, *last location*, *next page* and *exit* with respectively the following functions: jump to the table of contents, jump to the index, goto the last location in the document, goto next page and close the document. The labels to obvious destinations like *content* and *index* are predefined. Other predefined destinations are `FirstPage`, `LastPage`, `NextPage` and `PreviousPage`.

An action like `CloseDocument` is necessary to make an electronic document self containing. Other predefined actions you can use are `PrintDocument`, `SearchDocument` and `PreviousJump`. The meaning of these actions is obvious.



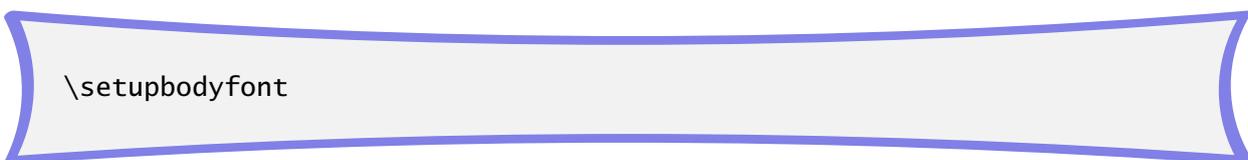
33.1 Introduction

The default font in `CONTEX`T is the *Computer Modern Roman* (`cmr`). In `CONTEX`T the following fonts are available.

For further reading we refer to the *Fonts in CONTEX*T manual where you can find information on how to install your own font.

33.2 Fontstyle and size

You can select the font family, style and size for a document with:



Name	Logical name	Also known as
Computer Modern Roman	cmr	Computer Modern Roman
Termes	termes	Times New Roman
Adventor	adventor	Avant Garde
Bonum	bonum	Bookman
Chorus	chorus	Zapf Chancery
Cursor	cursor	Courier
Heros	heros	Helvetica
Pagella	pagella	Palatino
Schola	schola	Century Schoolbook
Dejavu	dejavu	
Iwona	iwona	
Gentium	gentium	
Cambria	cambria	
Antykwa	antykwa	
Utopia	utopia	
LucidaBright	lucidanova	

Table 33.1 Fonts in `CONTEXT`.

If you typed `\setupbodyfont[chorus,9pt]` in the setup area of the input file your text would look something like this.

For changes in mid-document and on section level you should use:

`\swichtobodyfont`

On November 10th (one day before Saint Martinsday) the youth of Hasselt go from door to door to sing a special song and they accompany themselves on a `{\em foekepot}`. They won't leave before you give them some money or sweets. The song goes like this:

```
\startnarrower
\swichtobodyfont[heros,small]
\startlines
Foekepotterij, foekepotterij,
Geef mij een centje dan ga'k voorbij.
Geef mij een alfje dan blijf ik staan,
'k Zal nog liever naar m'n arrenmoeder gaan.
Hier woont zo'n rieke man, die zo vulle gèven kan.
Gèf wat, old wat, gèf die arme stumpers wat,
'k Eb zo lange met de foekepot elopen.
```

```
'k Eb gien geld om brood te kopen.
Foekepotterij, foekepotterij,
Geef mij een centje dan ga'k voorbij.
\stoptlines
\stopnarrower
```

Notice that `\start... \stopnarrower` is also used as a begin and end of the fontswitch. The function of `\start... \stoptlines` in this example is obvious.

On November 10th (one day before Saint Martinsday) the youth of Hasselt go from door to door to sing a special song and they accompany themselves on a *foekepot*. They won't leave before you give them some money or sweets. The song goes like this:

```
Foekepotterij, foekepotterij,
Geef mij een centje dan ga'k voorbij.
Geef mij een alfje dan blijf ik staan,
'k Zal nog liever naar m'n arrenmoeder gaan.
Hier woont zo'n rieke man, die zo vulle gèven kan.
Gèf wat, old wat, gèf die arme stumpers wat,
'k Eb zo lange met de foekepot elopen.
'k Eb gien geld om brood te kopen.
Foekepotterij, foekepotterij,
Geef mij een centje dan ga'k voorbij.
```

If you want an overview of the available font family you can type:

```
\showbodyfont[page]la]
```



	[pagella]										\mr : Ag		
	\tf	\sc	\sl	\it	\bf	\bs	\bi	\tfx	\tfixx	\tfa	\tfb	\tfc	\tfd
\rm	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag
\ss	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag
\tt	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag

33.3 Style and size switch in commands

In a number of commands one of the parameters is `style` to indicate the desired typestyle. For example:

```
\setuphead[chapter][style=\tfd]
```

In this case the character size for chapters is indicated with a command `\tfd`. But instead of a command you could use the predefined options that are related to the actual typeface:

```
normal bold slanted boldslanted type mediaeval
small smallbold smallslanted smallboldslanted smalltype
capital cap
```

33.4 Local font style and size

In the running text (local) you can change the *typestyle* into roman, sans serif and teletype with `\rm`, `\ss` and `\tt`.

You can change the *typeface* like italic and boldface with `\sl` and `\bf`.

The *typesize* is changed with `\switchtobodyfont`.

The actual style is indicated with `\tf`. If you want to change into a somewhat greater size you can type `\tfa`, `\tfb`, `\tfc` and `\tfd`. An addition of a, b, c and d to `\sl`, `\it` and `\bf` is also allowed.

```
{\tfc Mintage}
```

```
In the period from {\tt 1404} till {\tt 1585} Hasselt had its own
{\sl right of coinage}. This right was challenged by other cities,
but the {\switchtobodyfont[7pt] bishops of Utrecht} did not honour
these {\slb protests}.
```

The curly braces indicate begin and end of style or size switches.

Mintage

In the period from 1404 till 1585 Hasselt had its own *right of coinage*. This right was challenged by other cities, but the bishops of Utrecht did not honour these *protests*.

33.5 Redefining fontsize

For special purposes you can define your own size of the bodyfont.

```
\definebodyfont
```

A definition could look like this:

```
\definebodyfont[10pt][rm][tfe=Regular at 36pt]
{\tfe Hasselt!}
```

Now `\tfe` will produce 36pt characters saying:

Hasselt!

33.6 Small caps

Abbreviations like PDF () are printed in pseudo small caps. A small capital is somewhat smaller than the capital of the actual typeface. Pseudo small caps are produced with:

```
\cap
```

If you compare `\cap{hassel t}` and `\sc hassel t`: HASSELT and hasselt you can see the difference. The command `\sc` shows the real small caps. The reason for using pseudo small caps instead of real small caps is just a matter of taste.

33.7 Emphasized

To emphasize words consistently throughout your document you use:

`\em`

Emphasized words appear in a slanted style.

If you walk through Hasselt you should `{\bf \em watch out}` for `{\em Amsterdammers}`. An `{\em Amsterdammer}` is `{\bf \em not}` a person from Amsterdam but a little stone pillar used to separate sidewalk and road. A pedestrian should be protected by these `{\em Amsterdammers}` against cars but more often people get hurt from tripping over them.

This becomes:

If you walk through Hasselt you should ***watch out*** for *Amsterdammers*. An *Amsterdammer* is ***not*** a person from Amsterdam but a little stone pillar used to separate sidewalk and road. A pedestrian should be protected by these *Amsterdammers* against cars but more often people get hurt from tripping over them.

*An emphasize within an emphasize is normal again and a boldface emphasize looks like **this or this**.*

33

33.8 Teletype / verbatim

If you want to display typed text and want to keep your line breaking exactly as it is you use:

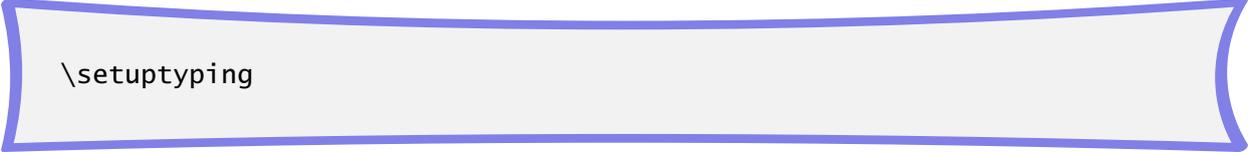
`\startTYPING`

In the text you can use:

`\type`

The curly braces enclose the text you want in teletype. You have to be careful with `\type` because the line breaking mechanism does not work anymore.

You can set up the 'typing' with:



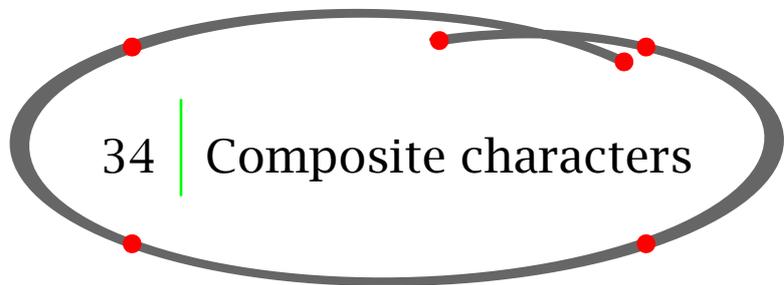
```
\setuptyping
```



```
\setuptype
```

33.9 Encodings

In $\text{CON}\text{T}\text{E}\text{X}\text{T}$ MKIV font encoding is no issue (anymore).



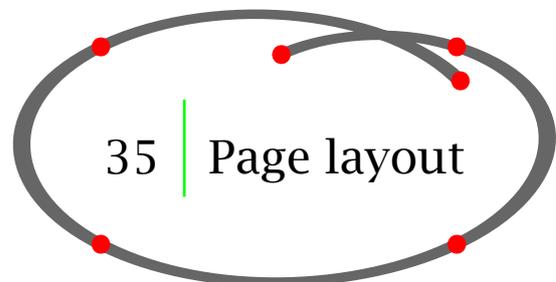
In chapter 3 you have already seen that you have to type more than one token to obtain special characters like # \$ % & _ { and }.

Characters with accents for example can be composed or coded with specific $\text{CON}\text{T}\text{E}\text{X}\text{T}$ commands in order to display them on paper. In case you have a text editor that can display utf8 you can type the composed characters directly.

It is not within the scope of this manual to go into accented characters in math mode. See the TEX Book by Donald E. Knuth on that subject.

Table 34.1 shows a few examples and the way you can code composed characters.

The character you want to display should be in the font.

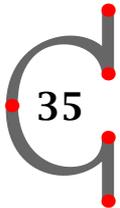


35.1 Introduction

The *Layouts in $\text{CON}\text{T}\text{E}\text{X}\text{T}$* manual by Willy Egger contains the necessary background information

Character	Composed	CON _T E _X T command	UTF8
ü	\"u	\uacute	ü
é	\'e	\egrave	é
â	\^a	\acircumflex	â
ä	\"a	\aacute	ä
à	\`a	\agrave	à
å	\aa	\aring	å
ç	\c{c}	\ccedilla	ç
ï	\"{\i}	\idiaeresis	ï
î	\^{i}	\icircumflex	î
Ä	\"A	\Adiaeresis	Ä
Å	\AA	\Aring	Å
É	\'E	\Egrave	É
æ	\ae	\aeligature	æ
Æ	\AE	\AEligature	Æ
ÿ	\"y	\ydiaeresis	ÿ

Table 34.1 Composed characters.



on page layout and design. Below you will find only the basic information necessary for defining rather simple layouts for paper and screen documents.

For more information (examples and usage) on the `\setuplayout` command please refer to the CON_TE_XT WIKI.

35.2 Designing the pagelayout

To be able to design a page layout you have to familiarize yourself with the pagemodel of CON_TE_XT. Figure 35.1 shows the areas on a page that you can use in your design.

The orange bodytext area contains the running text. The top, bottom, and edge area are useful for buttons in screen documents.

Please keep in mind that in CON_TE_XT you are defining/designing a right-hand page. Only after you have setup `\setuppagenumbering[alternative=doublesided]` the left page is available (mirrored right page).

Note in figure 35.2 that:

- the margintext (`\inmargin{m}`) is always in the left margin
- the footertext in the margin (`\setupfootertexts[margin][l][r][l]`) adapts automatically
- the page is completely mirrored when `alternative=doublesided`

Page layout

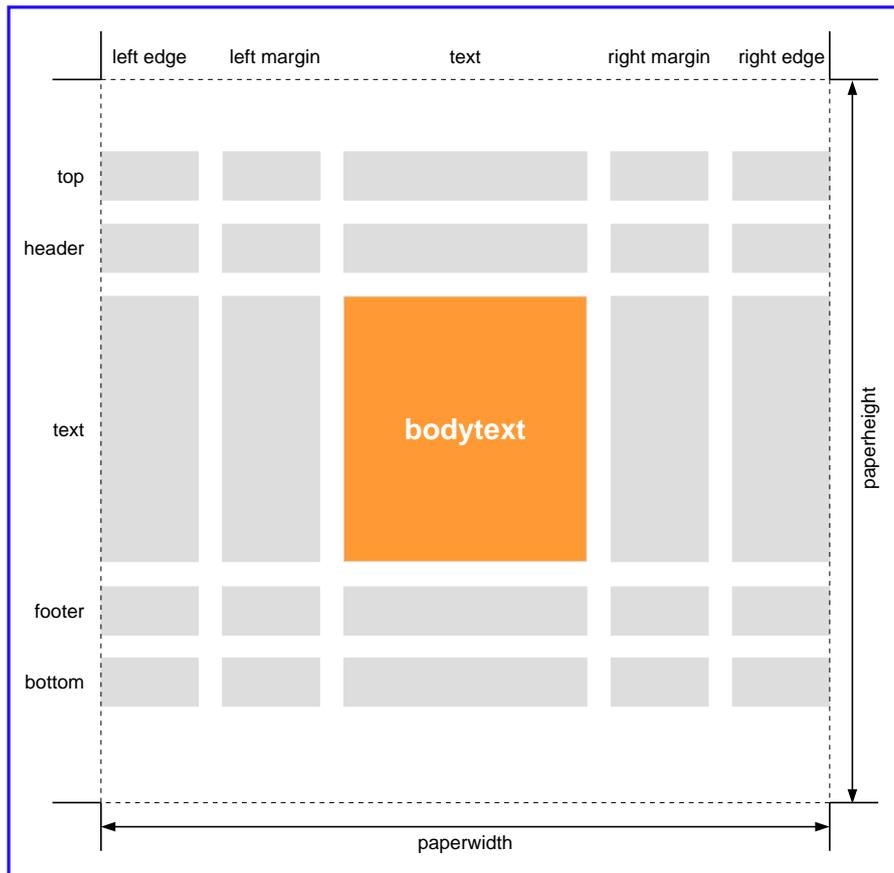


Figure 35.1 The page areas.

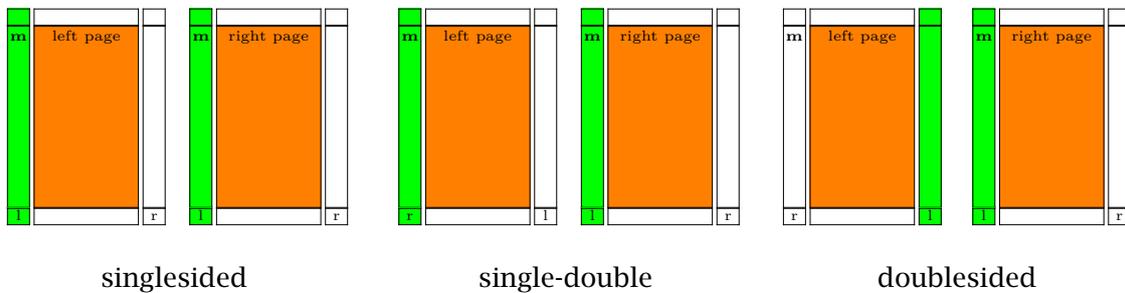


Figure 35.2 Page alternatives.

When designing a page ask yourself a few questions:

- do I want margin texts or margin figures
- will I use the margin for the section numbering
- do I have footer and/or header texts
- do I want a double sided layout (right-left page mirrored)
- do I use ornaments (like tabs) on the page
- do I have navigational buttons (screen documents)

35.3 Defining the papersize / screensize

Before you can set up your page layout you have to have an idea about the paper dimensions. The cutmarks connected by the dashed lines in figure 35.1 indicate the papersize. In `CONTEX`T you set up your papersize with:

Most common predefined papersizes in `CONTEX`T are A0..A10 and B1..B10 for paper and S3..S8 for screen documents.

Mostly you will use the default setup:

```
\setuppapersize
[A4][A4]
```

But you can also define your own paper size for specific products:

```
\definelayou
[postcard]
[width=15cm,
height=10cm]
```

35.4 Defining the page layout

The page layout is defined by:



```
\setuplayout
```

35

This command is typed in the set up area of your input file.

The layout of this manual was set with:

```
\setuplayout
[backspace=3cm,
margin=2cm,
margindistance=.5cm,
width=15cm,
topspace=2cm,
header=2cm,
footer=2cm,
height=25.7cm]
```

If you want to look at your page layout you can type the command `\showframe` and process one page or the whole file. The areas are shown in a number of frames.

The command `\showsetups` shows the values of the parameters. A combination of both commands is `\showlayout`.

The values of the layout parameters are available as commands. This enables you to work more accurately when defining measures of columns, figures and tables. A few of these parameters are explained in table 35.1.

Page layout

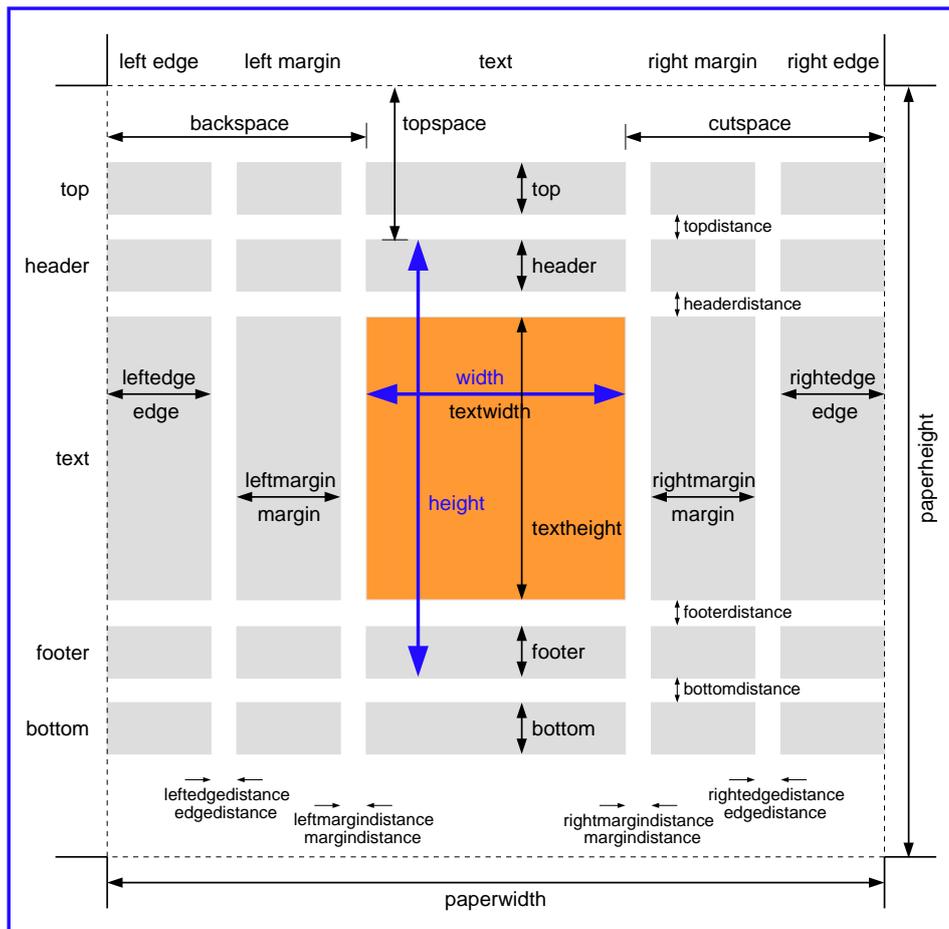


Figure 35.3 The page parameters.

Commands	Meaning
<code>\makeupwidth</code>	width of the typing area
<code>\makeupheight</code>	height of the typing area
<code>\textwidth</code>	width of the text area
<code>\textheight</code>	height of the text area

Table 35.1 A few parameters as commands.

If you want to define the width of a column or the height of a figure you can do it relative to the `\makeupwidth` or `\makeupheight`. Changes in this width or height will alter columns and figures proportionally.

```
\placefigure
[here]
[fig:stepgable]
{A stepgable.}
{\externalfigure[ma-cb-19][width=.6\textwidth]}
```

After processing this would become:



Figure 35.4 A steppable.

The other available values are (shown with `\showsetups`):

<code>\paperheight</code>	845.0468pt	29.7000cm	<code>\strutdepthfactor</code>	.28
<code>\paperwidth</code>	597.5079pt	21.0000cm	<code>\topskipfactor</code>	1.0
<code>\printpaperheight</code>	845.0468pt	29.7000cm	<code>\maxdepthfactor</code>	0.4
<code>\printpaperwidth</code>	597.5079pt	21.0000cm		
<code>\topspace</code>	42.6791pt	1.5000cm		
<code>\backspace</code>	64.0187pt	2.2500cm		
<code>\makeupheight</code>	759.6886pt	26.7000cm		
<code>\makeupwidth</code>	462.3573pt	16.2500cm		
<code>\topheight</code>	0ptpt	0.0000cm		
<code>\topdistance</code>	0ptpt	0.0000cm		
<code>\headerheight</code>	28.4527pt	1.0000cm		
<code>\headerdistance</code>	14.2264pt	0.5000cm		
<code>\textheight</code>	660.1040pt	23.2000cm		
<code>\footerdistance</code>	14.2264pt	0.5000cm		
<code>\footerheight</code>	42.6791pt	1.5000cm		
<code>\bottomdistance</code>	0ptpt	0.0000cm		
<code>\bottomheight</code>	0ptpt	0.0000cm		
<code>\leftedgewidth</code>	0ptpt	0.0000cm		
<code>\leftedgedistance</code>	0ptpt	0.0000cm		
<code>\leftmarginwidth</code>	56.9055pt	2.0000cm		
<code>\leftmargindistance</code>	14.2264pt	0.5000cm		
<code>\textwidth</code>	213.1787pt	7.4924cm		
<code>\rightmargindistance</code>	14.2264pt	0.5000cm		
<code>\rightmarginwidth</code>	56.9055pt	2.0000cm		
<code>\rightedgedistance</code>	0ptpt	0.0000cm		
<code>\rightedgewidth</code>	0ptpt	0.0000cm		
<code>\bodyfontsize</code>	8.0000pt	0.2812cm		
<code>\lineheight</code>	11.8720pt	0.4173cm		
<code>\strutheightfactor</code>	.72			

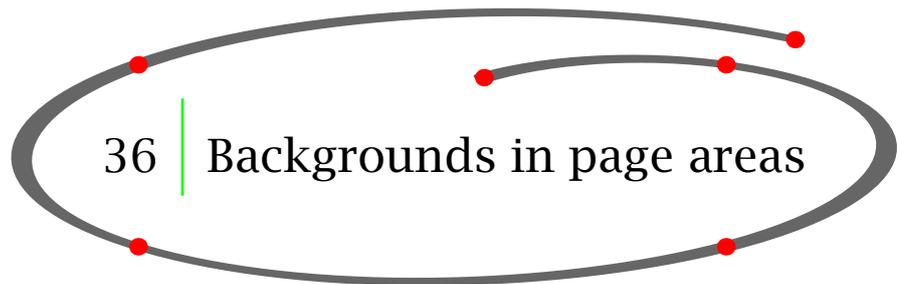
Backgrounds in page areas

The parameter values have a global effect and are default throughout the document. Nevertheless you might want to make slight changes in the page design for a number of pages.

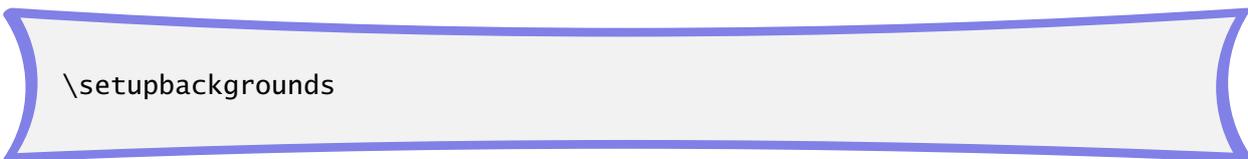
```
\adaptlayout[21,38][height=+.5cm]
```

In this case page 21 and 38 have a height of .5 cm + `textheight`.

It is advisable not to use these local changes too often. It is always better to alter the text than to change the page layout.



The page background can be set, with:



The first two bracket pairs are used to define the page areas. The last bracket pair is used for set up.

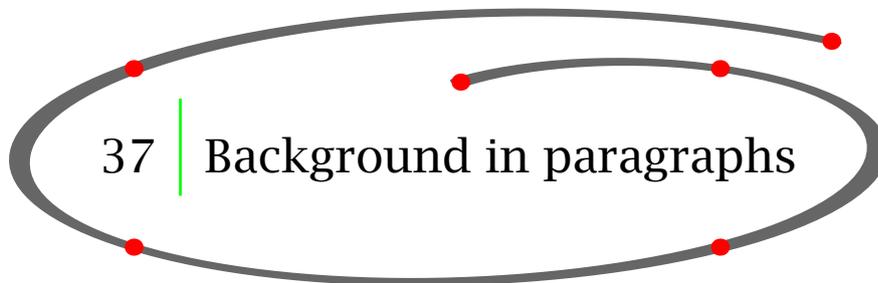
	left edge	left margin	text	right margin	right edge
top header					
text					
footer bottom					

Figure 36.1 The page areas defined in `\setupbackgrounds`.

If you want to have backgrounds in the gray areas of the page layout of figure 36.1 you type:

```
\setupbackgrounds
```

```
[header, text, footer]  
[leftmargin, text, rightmargin]  
[background=screen]
```



To emphasize a paragraph you can use backgrounds. A background is set with the command pair:

An example can illustrate the use:

```
\setuptextbackground  
[corner=round, frame=on,  
location=paragraph,  
leftoffset=.5\bodyfontsize,  
rightoffset=.5\bodyfontsize,  
bottomoffset=5pt]
```

```
\starttextbackground
```

Hasselt has produced a number of well known people. Only recently it turned out that Kilian van Rensselaer played a prominent role in the foundation of the State of New York.

```
\stoptextbackground
```

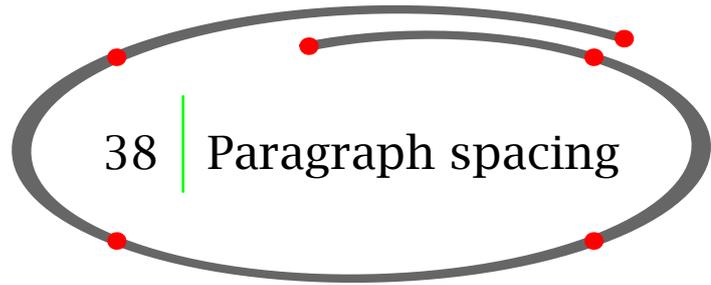
This would be displayed as:

Hasselt has produced a number of well known people. Only recently it turned out that Kilian van Rensselaer played a prominent role in the foundation of the State of New York.

Backgrounds can span multiple pages.

You can vary the display of the backgrounds with:

You can even define your own text backgrounds with:



38.1 Introduction

In $\text{T}_{\text{E}}\text{X}$ and $\text{CON}_{\text{T}}\text{E}_{\text{X}}\text{T}$ the most important unit of text is the paragraph. You can start a new paragraph by:

- an empty line
- the $\text{T}_{\text{E}}\text{X}$ command `\par`

In your ASCII input file you should use empty lines as paragraph separators. This will lead to a readable, clearly structured and well organized file and will prevent mistakes.

In situations where a command has to be closed explicitly you should use `\par`.

During one of the wars Hasselt lay under siege. After some time the city was famine stricken, everything edible was eaten. Except for one cow. The cow was kept alive and treated very well. `\par`

Once a day the citizens of Hasselt took the cow for a walk on the ramparts. The besiegers saw the well fed cow and became very discouraged. They broke up their camps and Hasselt was saved. `\par`

In the Hoogstraat in Hasselt there is a stone tablet with a representation of the cow that commemorates the siege and the shrewdness of the citizens of Hasselt.

This could also be typed without `\pars` and a few empty lines.

During one of the wars Hasselt lay under siege. After some time the city was famine stricken, everything edible was eaten. Except for one cow. The cow was kept alive and treated very well.

Once a day the citizens of Hasselt took the cow for a walk on the ramparts. The besiegers saw the well fed cow and became very discouraged. They broke up their camps and Hasselt was saved.

In the Hoogstraat in Hasselt there is a stone tablet with a representation of the cow that commemorates the siege and the wisdom of the citizens of Hasselt.



38.2 Inter paragraph spacing

The vertical spacing between paragraphs can be specified by:

```
\setupwhitespace
```

This document is produced with `\setupwhitespace[medium]`.

When inter paragraph spacing is specified there are two commands available that are seldom needed:

```
\nowhitespace  
\whitespace
```

When a paragraph consists of a horizontal line or a framed text like this:

```
Ridderstraat 27, 8061GH Hasselt
```

Sometimes spacing is suboptimal. For that purpose you could carry out a correction with:

```
\startlinecorrection
```

So if you would type:

```
\startlinecorrection  
\framed{Ridderstraat 27, 8061GH Hasselt}  
\stoplinecorrection
```

you will get a better output. Only use these commands if really needed!

```
Ridderstraat 27, 8061GH Hasselt
```

Another command to deal with vertical spacing is:

```
\blank
```

The bracket pair is optional and within the bracket pair you can type the amount of spacing. Keywords like `small`, `medium` and `big` are related to the fontsize.

In official writings Hasselt always has the affix `Ov`. This is an abbreviation for the province of `{\em Overijssel}`.

```
\blank[2*big]
```

The funny thing is that there is no other Hasselt in the Netherlands.

So it is redundant.

Paragraph spacing

`\blank`

The affix is a leftover from the times that the Netherlands and Belgium were one country under the reign of King Philip II of Spain.

`\blank[2*big]`

Hasselt in Belgium lies in the province of Limburg. One wonders if the Belgian people write Hasselt (Li) on their letters.

The command `\blank` without the bracket pair is the default space.

The example would become:

In official writings Hasselt always has the affix *Ov*. This is an abbreviation for the province of *Overijssel*.

The funny thing is that there is no other Hasselt in the Netherlands. So it is redundant.

The affix is a leftover from the times that the Netherlands and Belgium were one country under the reign of King Philip II of Spain.

Hasselt in Belgium lies in the province of Limburg. One wonders if the Belgian people write Hasselt (Li) on their letters.

The default spacing can be set up with:

`\setupblank`

If you want to surpress vertical spacing you can use:

`\startpacked`

In this manual the whitespace is set at medium. In the next situation this set up is ignored and the lines are packed.

`\startpacked`

Hasselt (Ov) lies in Overijssel.

Hasselt (Li) lies in Limburg.

Watch out: we talk about Limburg in Belgium. There is also a Dutch Limburg.

`\stoppacked`

This will become:

Hasselt (Ov) lies in Overijssel.

Hasselt (Li) lies in Limburg.

Watch out: we talk about Limburg in Belgium. There is also a Dutch Limburg.

It is not hard to imagine why there is also:

```
\startunpacked
```

You can force vertical space with `\godown`. The distance is specified within the brackets.

```
\godown
```

Try not to use this command. It is always better use the `\setup...` commands to setup your spacing model.

38.3 Whitespace before and after text components

Most text components that are coded with `CONTEXT` have a `\setup...` command with which you can define the whitespace before and after that component.

```
\setupitemize
  [before=,after=]
```

```
\setuphead
  [chapter]
  [before=,after=]
```

```
\setupframedtexts
  [before=,after=]
```

The use of the `\setup...` commands prevents you from having to code whitespaces throughout your `TEX` document. This would lead to unreadable sources and inconsistent use of whitespaces.

38.4 Skipping space

You can introduce horizontal and vertical space with `\hskip` and `\vskip` commands.

Try to avoid these commands in your text. It will probably lead to inconsistent spacing.

38.5 Indentation

You can set up the amount of the indentation with:

```
\setupindenting
```

A reasonable indentation is achieved by:

```
\setupindenting[yes,]
```

This will lead to indented paragraphs. By default, indentation after white space (as issued by `\blank`) is suppressed.

You can locally influence the indentation state by using:

```
\indenting
```

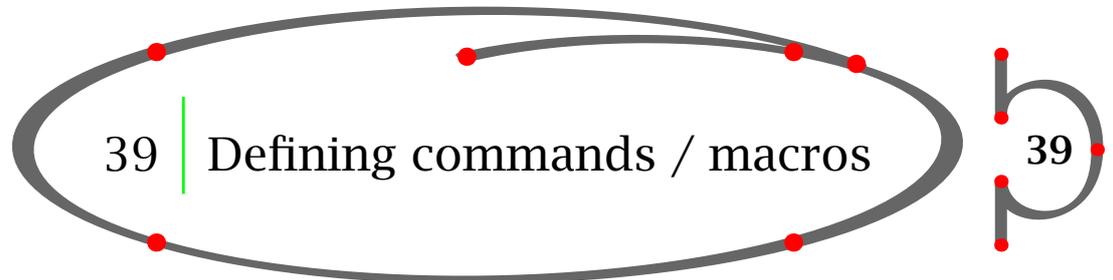
When for instance you say `never`, from that moment on indentation will be suppressed. Saying `none`, only influences the next paragraph.

If you choose to use indentations, and at a certain place you explicitly *do not* want to indent, you can also say:

```
\noindenting
```

In some `\setup...` commands you can set the parameter `indent=yes`. This means that the paragraph that follows the textcomponent will indent:

```
\setupitemize[indentnext=yes]
```



CONTEXT is a set of macros based on T_EX. T_EX is a programming language as well as a typographical system. This means that you can do the programming yourself if you need that kind of flexibility.

You can define a new command with:

```
\define
```

The next example will explain its meaning.

You may have a well illustrated document and you are tired of typing:

```
\placefigure
```

```
[here, force]
[fig:logical name]
{Caption.}
{\externalfigure[filename] [width=5cm]}
```

You could define your own command with a few variables like:

- logical name
- caption
- file name

Your command definition and call could look something like this:

```
\define[3]\myputfigure
  {\placefigure
   [here, force] [fig:#1]
   {#2}{\externalfigure[#3] [width=5cm]}}
\myputfigure{lion}{The Dutch lion is a sentry.}{ma-cb-13}
```

From then on the `\myputfigure` is available. Between brackets [3] indicates that you want to use three variables #1, #2 and #3. In the command call `\myputfigure` you have to place these variables between curly braces. The result is shown in figure 39.1.



Figure 39.1 The Dutch lion is a sentry.

Very sophisticated commands can be programmed, but this is left to your own inventiveness.

In addition to defining commands you can also define `\start... \stop` command pairs.

`\definestartstop`

For example:

```

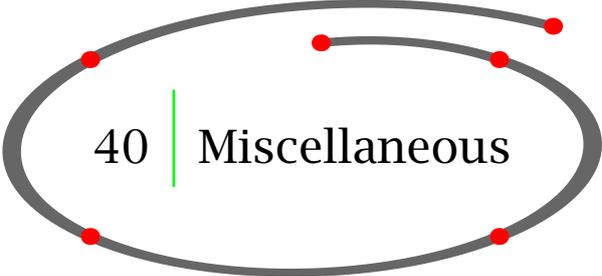
\definestartstop
  [attention]
  [before=\blank\startmarginrule,
   after=\stopmarginrule\blank]

\startattention
{\em Hasselter Juffers} are sweet cookies but the name is no
coincidence. On July 21 in 1233 the {\em Zwartewaterklooster}
(Blackwater Monastery) was founded. The monastery was meant
for unmarried girls and women belonging to the nobility of
Hasselt. These girls and women were called {\em juffers}.
\stopattention

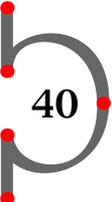
```

This will result in:

Hasselter Juffers are sweet cookies but the name is no coincidence. On July 21 in 1233 the *Zwartewaterklooster* (Blackwater Monastery) was founded. The monastery was meant for unmarried girls and women belonging to the nobility of Hasselt. These girls and women were called *juffers*.



40 | Miscellaneous



40

40.1 A titlepage

In the first example of this manual on page 5 we used the command:

```
\startNAMEmakeup
```

This command can be used to define titlepages. Such a command is needed since title pages often have a different layout than that of the bodytext. With the command pair `\start ... \stopstandardmakeup` you can make up a page within the default page dimensions.

A simple titlepage may look like this:

```

\startstandardmakeup
\blank
\rightaligned{\tfd Hasselt in the 21st century}
\blank

```

```

\rightaligned{\tfb The future}
\vfill
\rightaligned{\tfa C. van Marle}
\rightaligned{Hasselt, 2013}
\stopstandardmakeup

```

In a doublesided document you have to go through some additional actions to typeset the back of the titlepage.

```

\startstandardmakeup[doublesided=no]
\blank
\rightaligned{\tfd Hasselt in the 21st century}
\blank
\rightaligned{\tfb The future}
\vfill
\rightaligned{\tfa C. van Marle}
\rightaligned{Hasselt, \currentdate[year]}
\stopstandardmakeup
\startstandardmakeup[page=no]
\vfill
\copyright \currentdate[year]

```

This book is dedicated to the people living in Hasselt. We want to thank photographer J. Jonker for manipulating the photos in this book in such a way that readers can get a clear picture of Hasselt's future look.

```
\stopstandardmakeup
```

Your own make ups can be made and set up with:

40

```
\definemakeup
```

and

```
\setupmakeup
```

Please refer to the `CONTEXT` WIKI for more information on the `\start... \stopmakeup` command.

40.2 Overlays

The overlay mechanism gives you the opportunity to add a specific layout to a text component. When there is a background option in a `CONTEXT` command you can use overlays.

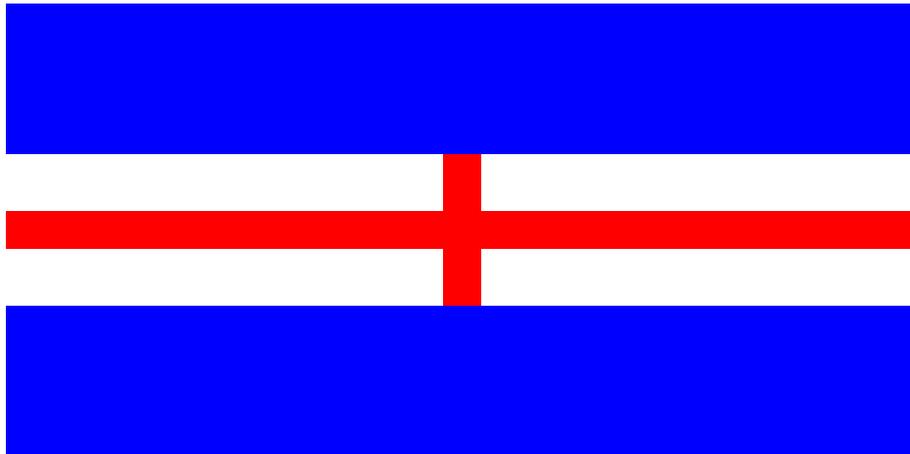
The flag of Hasselt could be defined with framed and a number of overlays:

```
\defineoverlay
  [verticalbar]
  [{\blackrule[height=2cm,width=.5cm,color=red]}]

\defineoverlay
  [horizontalbar]
  [{\blackrule[height=.5cm,width=12cm,color=red]}]

\framed
  [width=12cm,
   height=6cm,
   background={color,foreground,verticalbar,horizontalbar},
   offset=overlay,
   backgroundcolor=blue,
   frame=off]
  {\blackrule[width=12cm,height=2cm,color=white]}
```

This will become:



The pagenumber in this manual has a background with an overlay where the `\MPClipFive` command takes care of drawing the image with METAPOST.

```
\defineoverlay
  [NumberBackground]
  [{\MPClipFive{\overlaywidth}{\overlayheight}{30pt}{5pt}]

\setuppagenumbering
  [\location={footer,middle},
   \command=\NummerCommand]

\def\NummerCommand#1%
  {\framed
   [\background=NumberBackground,
    \frame=off,
    \offset=6pt]
   {\lower.5\dp\strutbox\hbox spread 60pt{\hss#1\hss}}}
```

40.3 Setups

While defining the layout of a document you can define setups with `\start... \stopsetups`. Setups are placed in the setup area of input file and mostly used to combine a number of commands.

```
\startsetups colorize
  \blue
\stopsetups
\startsetups decolorize
  \black
\stopsetups
\setupitemize
  [before=\setups{colorize},
   after=\setups{decolorize}]
```

Some data on the church are:

```
\startitemize[packed,3*broad]
\sym{997} mentioned for the first time
\sym{1380} destroyed by fire
\sym{1466} rebuild
\sym{1657} restored after shelling by enemy troops
\sym{1725} struck by lightning
\stopitemize
```

Which would result in:

Some data on the church are:
 997 mentioned for the first time
 1380 destroyed by fire
 1466 rebuild
 1657 restored after shelling by enemy troops
 1725 struck by lightning

40

Another way of invoking the setups is by the `setups` option that comes with some `CONTEX`T commands:

```
\definesstartstop[remark]
\setupstartstop[remark]
  [before=\startframed,
   after=\stopframed]
\startsetups important
  \inleftmargin
  [scope=local,
   hoffset=1em]{\bf\color[blue]{} }
\stopsetups
\setupframed
```

```
[align=normal,
 setups=important,
 frame=on,
 framecolor=blue,
 offset=5pt]

\startremark
The Stephanus Church was built in 997. After an enormous
fire in 1380 it was rebuilt and that's why it has Gothic
features. The rebuilding was finished in 1466.\endgraf
\stopremark
```

This becomes:

The Stephanus Church was built in 997. After an enormous fire in 1380 it was rebuilt and that's why it has Gothic features. The rebuilding was finished in 1466.

40.4 Variables

There is a mechanism in `CONTEX`T that enables you to compact information in a list of variables that you can recall throughout the document.

The example below shows how to use variables in defining a coverpage.

```
\setvariables
[cover]
[set=\setups{coverpage},
 student=no,
 teacher=yes,
 title=From Hasselt to America,
 subtitle=An Odyssey,
 authors=\setup{allauthors},
 edition=2012,
 isbn=0123456789]
```

The moment you need the title on your cover page (or somewhere else in your document) you can summon it by:

```
\getvariable{cover}{title}
```

40.5 Floating blocks

A block in `CONTEX`T is a text element, for example a table or a figure that you can process in a special way. You have already seen the use of `\placefigure` and `\placetable`. These are both examples of floating blocks. The floating mechanism is described in chapter 12 and 13.

You can define these kind of blocks yourself with:

```
\definefloat
```

The bracket pairs are used for the name in singular and plural form. For example:

```
\definefloat[intermezzo][intermezzi]
```

Now the following commands are available:

```
\placeintermezzo[] [] {} {}
\startintermezzotext ... \stopintermezzotext
\placelistofintermezzi
\completelistofintermezzi
```

The newly defined floating block can be set up with:

```
\setupfloat
```

You can set up the layout of floating blocks with:

```
\setupfloats
```

You can set up the numbering and the labels with:

40

```
\setupcaption
```

These commands are typed in the set up area of your input file and will have a global effect on all floating blocks.

```
\setupfloat[intermezzo][location=middle]
\setupcaption[location=bottom,headstyle=boldslanted]
```

```
\placeintermezzo{An intermezzo.}
\startframedtext
```

At the beginning of this century there was a tram line from Zwolle to Blokzijl via Hasselt. Other means of transport became more important and just before the second world war the tram line was stopped. Nowadays such a tram line would have been very profitable.

```
\stopframedtext
```

At the beginning of this century there was a tram line from Zwolle to Blokzijl via Hasselt. Other means of transport became more important and just before the second world war the tram line was stopped. Nowadays such a tram line would have been very profitable.

Intermezzo 40.1 An intermezzo.

The framed texts inherits its layout from the example page 55.

Tables or figures may take up a lot of space. The placing of these text elements can be postponed till the next page break. This is done with: `\start ... \stoppostponing`:

```
\startpostponing
\placefigure
  {A postponed figure.}
  {\externalfigure[ma-cb-16][width=\textwidth]}
\stoppostponing
```

The figure will be placed at the top of the next page and will cause minimal disruption of the running text.

40.6 Storing text for later use

You can store information temporarily for future use in your document with:

`\startBUFFER`

`\startbuffer`

40

For example:

```
\startbuffer[visit]
If you want to see what Hasselt has in store you should come and
visit it some time. If you take this manual with you, you will
recognise some locations.
\stopbuffer

\getbuffer[visit]
```

With `\getbuffer[visit]` you recall the stored text. The logical name is optional. With `\typebuffer[visit]` you get back the typeset version of the content of the buffer.

Buffers are set up with:



Figure 40.1 A postponed figure.

40

`\setupbuffer`

You can also save a buffer to an external file with:

If you want to save the buffer `visit` in an external file called `myfile-sightseeing.tmp` you type:

`\savebuffer[visit][sightseeing]`

40.7 Lines

There are many comands to draw lines. For a single line you type:

`\hairline`

or:

```
\thinrule
```

For more lines you type:

```
\thinrules
```

Text in combination with lines is also possible:

— **Hasselt - Amsterdam** —

If you draw a straight line from Hasselt to Amsterdam you would have to cover a distance of almost 145 km.

If you draw two straight lines from Hasselt to Amsterdam you would have to cover a distance of almost 290 km.

Amsterdam _____

_____ Hasselt

The code of this example is:

```
\starttextrule{Hasselt -- Amsterdam}
If you draw a straight line from Hasselt to Amsterdam you would have
to cover a distance of almost 145 \unit{Kilo Meter}.
\stoptextrule

If you draw two straight lines from Hasselt to Amsterdam you would
have to cover a distance of almost 290 \unit{Kilo Meter}.

Amsterdam \thinrules[n=3] Hasselt
```

You always have to be careful in drawing lines. Empty lines around `\thinrules` must not be forgotten and the vertical spacing is always a point of concern.

You can set up line spacing with:

```
\setupthinrules
```

There are a few complementary commands that might be very useful.

```
\setupfillinrules
```

These commands are introduced in the examples below:

```
\setupfillinrules[width=2cm]
\setupfillinlines[width=3cm]
\fillinrules[n=1]{\bf name}
\fillinrules[n=3]{\bf adress}
\fillinlines{Can you please state the \underbar{number} of houses
in Hasselt.} \par
Strike out \overstrikes{Hasselt in this text}\periods[18]
```

This will become:

name _____
adress _____

Can you please state the number of houses in Hasselt. _____

Strike out Hasselt in this text.....

These commands are used in questionnaires. Text that is struck out or underlined will not be hyphenated.

In section 40.2 you have already seen the use of the \blackrule command that can be set up with:

40

```
\setupblackrules
```

```
\blank
\blackrule[width=\textwidth,height=1cm,color=blue]
```

This will result in a rather fat line:



40.8 Super- and subscript in text

Hasselt's economy has known its ^{ups} and _{downs}. Since the nineties of the last century its economy is ^{so}_{so}.

This ugly text was made with `\low{}`, `\high{}` and `\lohi{ }{ }`. The text was placed between the curly braces.

40.9 Date

You can invoke the system date in your text with:

```
\currentdate
```

With `\currentdate[day]`, `\currentdate[month]` and `\currentdate[year]` you can invoke day, month and year separately.

40.10 Rotating text

Sometimes you may want to rotate text or images. You can rotate text and other objects with:

```
\rotate
```

The first bracket pair is optional. Within that bracket pair you specify the rotation: `rotation=90`. The curly braces contain the text or object you want to rotate.

Hasselt got its municipal rights in 1252. From that time on it had the `\rotate[rotation=90]{right}` to use its own seal on official documents. This seal showed Holy Stephanus known as one of the first Christian martyrs, and was the `\rotate[rotation=270]{patron}` of Hasselt. After the Reformation the seal was redesigned and Stephanus lost his `\quote{holiness}` and was from that time on depicted without his aureole.

This results in a very ugly paragraph:

Hasselt got its municipal rights in 1252. From that time on it had the `right` to use its own seal on official documents. This seal showed Holy Stephanus known as one of the first Christian martyrs, and was the `patron` of Hasselt. After the Reformation the seal was redesigned and Stephanus lost his 'holiness' and was from that time on depicted without his aureole.

You can rotate an image just as easily:

```
\placefigure
  [] [fig:rotation]
  {The 180 \unit{Degrees} rotated fishing port (de Vispoort).}
  {\rotate[rotation=180]{\externalfigure[ma-cb-15][width=10cm]}}
```

You can see in figure 40.2 that it is not always clear what you get when you rotate.



Figure 40.2 The 180° rotated fishing port (de Vispoort).

We can set up rotating with:

```
\setuprotate
```

In the example above you could also rotate image and caption by:

```
\placefigure
  [180][fig:rotation]
  {The 180 \unit{Degrees} rotated fishing port (de Vispoort).}
  {\externalfigure[ma-cb-15][width=10cm]}
```

40

40.11 Scaling text

For some obscure reasons you may want to scale text. You can scale text and other objects with:

```
\scale
```

After 1810 the Dedemsvaart brought some prosperity to Hasselt. All ships went through the canals of Hasselt and the `\scale[factor=10]{shops}` on both sides of the canals `\scale[factor=10]{prospered}`.

Which will result in:

After 1810 the Dedemsvaart brought some prosperity to Hasselt. All ships went through the canals of Hasselt and the **shops** on both sides of the canals **prospered**.

`\stoptlines`

This little rhyme contains a warning for the magistrates of Hasselt: don't allow personal benefits or feelings to influence your wisdom in decision making.

This will become:

On a wooden panel in the town hall of Hasselt you can read:

Heimelijcken haet
eigen baet
jongen raet
Door diese drie wilt verstaen
is het Roomsche Rijck vergaen.

This little rhyme contains a warning for the magistrates of Hasselt: don't allow personal benefits or feelings to influence your wisdom in decision making.

In a few commands new lines are generated by `\\`. For example if you type `\inmargin{in the\\margin}` then the text will be divided over two lines.

40.14 Hyphenation

When writing multi-lingual texts you have to be aware of the fact that hyphenation may differ from one language to another.

To activate a language you type:

`\mainlanguage`

40

Between the brackets you fill in `af`, `ca`, `cs`, `cs`, `da`, `de`, `en`, `fi`, `fr`, `it`, `la`, `nl`, `nb`, `nn`, `pl`, `pt`, `es`, `sv` and `tr` for afrikaans, catalan, czech, slovak, danish, german, english, finnish, french, italian, latin, dutch, bokmal, nynorsk, polish, portuguese, spanish, swedish and turkish respectively.

To change from one language to another you can use:

`\language[nl] \language[en] \language[de] \language[fr] \language[sp] ...`

or the shorthand versions:

`\nl \en \de \fr \sp ...`

An example:

If you want to know more about Hasselt, the best book to read is probably `\quote{nl Uit de geschiedenis van Hasselt}` by F.~Peereboom.

If you want to know more about Hasselt, the best book to read is probably 'Uit de geschiedenis van Hasselt' by F. Peereboom.

If a word is wrongly hyphenated you can define the hyphenation points yourself. This is done in the set up area of your input file:

```
\hyphenation{his-to-ry}
```

Note that the language setting is also responsible for the way quotes are placed around quotes and quotations (see section 17).

In some languages (like Dutch) compound words are used that are connected with a hyphen. The separate words have to be hyphenated correctly. In order to do that you can use ||.

If your looking for an English||speaking person in Hasselt you should go to the Tourist Information Office. There you may expect to find full|| and part||time employees who are fluent in German, English, French and of course Dutch.

This will become:

If your looking for an English-speaking person in Hasselt you should go to the Tourist Information Office. There you may expect to find full- and part-time employees who are fluent in German, English, French and of course Dutch.

The double || takes care of the hyphen and the correct hyphenation of the separate words. Also note the suspended compounds.

40.15 Charts

To enable you to draw flow diagrams `CONTEXT` contains the core module chart. A simple organogram may look like this:



This diagram is defined with the commands below:

```

\setupFLOWcharts
  [width=9\bodyfontsize,
   height=2\bodyfontsize,
   dx=1\bodyfontsize,
   dy=1\bodyfontsize]

\setupFLOWlines
  [arrow=no]

\startFLOWchart[organogram]
  \startFLOWcell
    \shape {action}
    \name {01}
    \location {2,1}
  
```

```

\text      {Zwartewaterland}
\connect  [bt]{02}
\connect  [bt]{03}
\connect  [bt]{04}
\stopFLOWcell
\startFLOWcell
\shape    {action}
\name     {02}
\location {1,2}
\text     {Hasselt}
\stopFLOWcell
\startFLOWcell
\shape    {action}
\name     {03}
\location {2,2}
\text     {Zwartsuis}
\stopFLOWcell
\startFLOWcell
\shape    {action}
\name     {04}
\location {3,2}
\text     {Genemuiden}
\stopFLOWcell
\stopFLOWchart

```

It is of good practice to define your setups and flow diagrams in separate definition files (environments).

The flowchart can then be invoked by:

```
\FLOWchart[organogram]
```

40

40.16 Comment in input file

All text between `\start...` `\stoptext` will be processed while running `CONTEXT`. Sometimes however you may have text fragments you don't want to be processed or you want to comment on your `CONTEXT` commands.

If you precede your text with the percentage sign `%` it will not be processed.

```

% In very big documents you can use the command \input for
% different files.
%
% For example:
%
% \input hass01.tex % chapter 1 on Hasselt
% \input hass02.tex % chapter 2 on Hasselt
% \input hass03.tex % chapter 3 on Hasselt

```

When you delete the % before `\input` the three files will be processed. The comment describing the contents of the files will not be processed.

40.17 Notes

If you want your comment in the input file visible as a 'note' in the PDF file you can use:

```
\startcomment
```

```
\startcomment
```

```
  The image of the Vispoort should be in color.
```

```
\stopcomment
```

The command will produce a sticky note in the PDF.

The note is only visible when interactivity is set with `\setupinteraction` and the comment with `\setupcomment`.

40.18 Hiding text

Text can be hidden with:

```
\starthiding
```

The text between `\start ... \stophiding` will not be processed.

40.19 Input of another tex file

In a number of situations you may want to insert other T_EX files in your input file. For example, sometimes it is more efficient to specify CON_TE_XT sources in more than one file in order to be able to partially process your files.

Another file (with the name `another.tex`) can be inserted by:

```
\input another.tex
```

The extension is optional so this will work too:

```
\input another
```

The command `\input` is a T_EX command.

For a more systematic approach in maintaining your documents CON_TE_XT supports a project structure with commands like `\start... \stopenvironment` and `\start... \stopproduct`. Please refer to the magazine *Project structure* for more information.

40.20 XML (eXtended Markup Language)

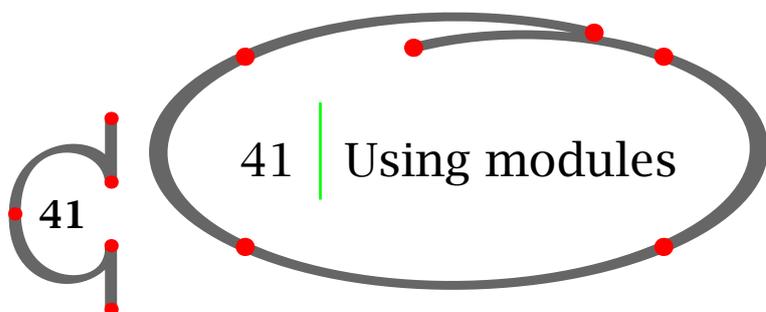
Normally you code your document with CONTEX T commands so you can tell CONTEX T what to do with the coded text elements.

A more rigid way to code your content is XML (eXtended Markup Language) which enables you to have more control over your content (scripting, xslt, validation). A simple XML coded document could look like this:

```
<?xml version='1.0' standalone='yes?'>
<document>
  <section>
    <title>Hasselt in winter</title>
    <content>
      <p>In winter scating is a very popular sport in Hasselt.
        All over Hasselt the frozen canals offer children a great
        play ground.</p>
      <p>...</p>
    </content>
  </section>
</document>
```

CONTEX T is able to deal with XML directly without underlying XML2TEX conversions. Please refer to the manual Dealing with XML for more information on how to process XML documents.

CONTEX T also supports MATHML (presentational and content markup) and OPENMATH with which math expressions can be coded in XML documents.



For reasons of efficiency CONTEX T comes with a number of modules that contain specific functionality. Loading a module is done in the set up area of your input file by means of:

```
\usemodule
```

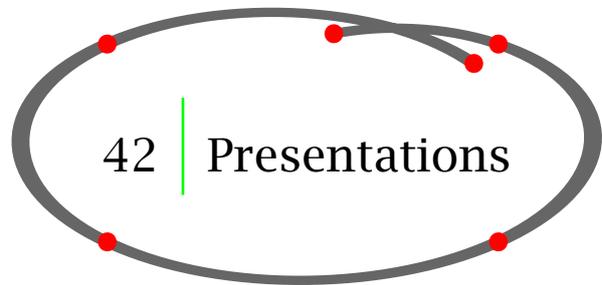
When you load a module CONTEX T looks for a file with the following (prefix-)name:

- m-modulename (core module)
- p-modulename (private module)

- s-modulename (CONTEX style file)
- x-modulename (XML module)
- t-modulename (third party module)
- modulename

A few example core modules are:

- m-fields (m-fields.mkiv): for PDF forms
- m-morse (m-morse.mkiv): for morse
- m-spreadsheet (m-spreadsheet.mkiv): for spreadsheets
- m-visual (m-visual.mkiv): for visual debugging
- m-zint (m-zint.mkiv): for generating bar codes
- s-pre-** (s-pre-**): for presentations



You can use CONTEX for making your own presentations. A CONTEX presentation is an interactive PDF document with a screen layout. Often presentations are good examples of the cooperation between CONTEX and METAPOST.

CONTEX comes with a number ready-to-use presentations. A presentation is a module with the prefix s- and that you can load with the `\usemodule` command.

If you want to use an already existing presentation the best way to proceed is:

- goto `../your-contextdir/tex/texmf-context/tex/context/base` in your text editor
- open a presentation: for example `s-pre-05.tex`
- goto the end of the file and study the commands between the `\start... \stoptext` pair
- copy the commands into your own presentation file
- invoke the presentation with `\usemodule[s][pre-05]` in the setup area of your presentation file
- process the file to view the result
- edit the content of your presentation

A stepwise setup of a presentation is given at the CONTEX WIKI.



43 | Graphical extension / METAPOST

The graphical possibilities of $\text{T}_{\text{E}}\text{X}$ -related macro packages are rather limited. However, by using the graphical package METAPOST of John Hobby a complete range of graphical features has become available that may improve the look of your documents.

In $\text{CON}_{\text{T}}\text{E}_{\text{X}}\text{T}$ there is a direct link to METAPOST so users can apply the features of METAPOST directly into their documents. The chapter headers and page numbers of this manual are extended by some graphical elements that are generated by METAPOST.

If you look carefully at these METAPOST extensions you will notice a lot of contextual adaptation (width and height dependend) and randomization. So you can do things in your document that are not possible in other typesetting applications.

A more practical example (for a mathematician at least) is drawn in figure 43.1:

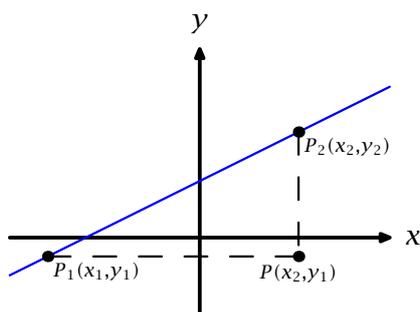


Figure 43.1
METAPOST example.

43

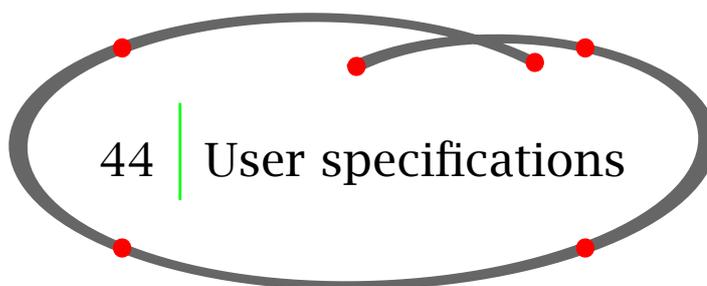
This example is taken from the mathematical text book *Algetrigulus* by Philip Brown. All graphics in his book are made by means of METAPOST. This one is defined by:

```
\startreusableMPgraphic{origin}
  path pb; pb:=(5.5cm,0cm)..(10.5cm,0cm);
  path qb; qb:=(8cm,-1cm)..(8cm,2.5cm);
  pickup pencircle scaled 0.5mm;
  drawarrow pb;
  drawarrow qb;
  draw thelabel.rt(btex $x$ etex,(10.6cm,0cm));
  draw thelabel.top(btex $y$ etex,(8cm,2.6cm));
  path l; l:=(5.5cm,-0.5cm)..(10.5cm,2cm);
  pickup pencircle scaled 0.3mm;
  draw l withcolor blue ;
```

User specifications

```
pair A; A:=(6cm,-0.25cm);
pair B; B:=(9.3cm,1.4cm);
pair C; C:=(9.3cm,-0.25cm);
pickup pencircle scaled 0.15cm;
drawdot A; drawdot B; drawdot C;
draw thelabel.lrt(btex $\scriptstyle P_1(x_1,y_1)$ etex ,A);
draw thelabel.lrt(btex $\scriptstyle P_2(x_2,y_2)$ etex ,B);
draw thelabel.bot(btex $\scriptstyle P(x_2,y_1)$ etex ,C);
path s; s:=A..(9.3cm,-0.25cm);
draw s dashed (evenly scaled 1mm) withpen pencircle scaled 0.3mm;
path t; t:=B..(9.3cm,-0.25cm);
draw t dashed (evenly scaled 1mm) withpen pencircle scaled 0.3mm;
\stoptreusableMPgraphic
```

The usage and features of METAPOST within CONTEX T are described in the extensive METAFUN manual.



The setup area of your document is the area before the `\starttext` command. For example:

<code>\setuplayout[width=25cm]</code>	first line of your file set the width of your text empty line for readability
<code>\starttext</code>	starts your text
Hello Hasselt.	your text
<code>\stoptext</code>	ends your text

Note that the first line of this file is empty. However, this first line is a preamble and can be used for specific user specifications. For example:

<code>% engine=luatex</code>	use the <code>luatex</code> engine
	empty line for readability
<code>\setuplayout[width=25cm]</code>	set the width of your text
	empty line for readability
<code>\starttext</code>	starts your text
Hello Hasselt.	your text
<code>\stoptext</code>	ends your text

Note that CONTEX T sees the text after the `%` sign in this first line not as a comment.



User specifications

The preamble can have a meaning for both `CONTEXT` and `SCITE`:

```
% engine=pdftex interface=en modes=screen language=uk  
  
\starttext  
Hello Hasselt.  
  
\stoptext
```

This will be interpreted as:

<code>engine=pdftex</code>	<code>CONTEXT</code> : run as <code>PDFTEX</code>
<code>interface=en</code>	<code>CONTEXT</code> : expect english <code>CONTEXT</code> commands (lexing) <code>SCITE</code> : use english lexing
<code>modes=screen</code>	<code>CONTEXT</code> : invoke mode <code>screen</code> that is set in the text
<code>language=uk</code>	<code>SCITE</code> : use the english spell checker

A | Command definitions

Here we summarize the commands we introduced in the previous chapters. This is just a selection of the whole repertoire of $\text{CON}\text{T}\text{E}\text{X}\text{T}$ commands. Those who want to see them all can take a look at the more extensive manual or the *Quick Reference Manuals* that give a complete overview of all $\text{CON}\text{T}\text{E}\text{X}\text{T}$ -commands.

Arguments that are typeset *slanted* are optional and can be omitted. The number points to the page where the command is explained. Black arrows indicate that the command is only of use in interactive documents and gray arrows tell us that additional functionality is provided in interactive mode. Keep in mind that we only show the commands we described in this manual, there are many more.

<p><code>\about {¹.}</code> [<code>².</code>]</p> <p>1 CONTENT 2 REFERENCE</p>	<p><code>\cap {[*].}</code></p> <p>* CONTENT</p>
<p><code>\at {¹.}</code> [<code>².</code>] [<code>³.</code>]</p> <p>1 CONTENT 2 CONTENT 3 REFERENCE</p>	<p><code>\chapter [...¹.]</code> [<code>².</code>]</p> <p>1 REFERENCE 2 CONTENT</p>
<p><code>\blank [...[*].]</code></p> <p>* small medium <u>big</u> nowhite back white disable force reset line halfline FORMULA fixed flexible none always outer joinedup</p>	<p><code>\color [¹.]</code> [<code>².</code>]</p> <p>1 TEXT 2 CONTENT</p>
<p><code>\button [...¹.]</code> [<code>².</code>] [<code>³.</code>]</p> <p>1 <i>inherits from</i> <code>\setupbuttons</code> 2 CONTENT 3 REFERENCE</p>	<p><code>\crlf</code></p>

Command definitions

`\currentdate [...,*,...]`

* *inherits from* \date

`\define [..1.] \..2. {..3.}`
OPT

- 1 NUMBER
- 2 CSNAME
- 3 CONTENT

`\definebodyfont [...1,...] [2.]
[...3,,...]` OPT

- 1 5pt ... 12pt small big
- 2 rm ss tt hw cg mm
- 3 tf = FILE
bf = FILE
sl = FILE
it = FILE
bs = FILE
bi = FILE
sc = FILE
mr = FILE
ex = FILE
mi = FILE
sy = FILE
ma = FILE
mb = FILE
mc = FILE
md = FILE

`\definecolor [..1.] [...2,,...]`

- 1 IDENTIFIER
- 2 r = TEXT
g = TEXT
b = TEXT
c = TEXT
m = TEXT
y = TEXT
k = TEXT
s = TEXT
h = TEXT
t = TEXT
a = TEXT
p = TEXT
e = TEXT

`\definecombinedlist [..1.] [...2,...]
[...3,,...]`
OPT

- 1 IDENTIFIER
- 2 LIST
- 3 *inherits from* \setupcombinedlist

`\definedescription [..1.] [...2,,...]`
OPT

- 1 IDENTIFIER
- 2 *inherits from* \setupdescriptions

`\defineenumeration [...1,...] [2.]
[...3,,...]` OPT

- 1 IDENTIFIER
- 2 IDENTIFIER
- 3 *inherits from* \setupenumerations

`\definefloat [..1.] [2.]`

- 1 SINGULAR NAME
- 2 PLURAL NAME

`\definehead [..1.] [2.]`

- 1 IDENTIFIER
- 2 SECTION

`\definemakeup [..1.] [...2,,...]`

- 1 IDENTIFIER
- 2 *inherits from* \setupmakeup

`\defineregister [..1.] [2.]`

- 1 SINGULAR NAME
- 2 PLURAL NAME

Command definitions

`\definesorting` [^{1.}] [^{2.}] [^{3.}]
OPT

- 1 SINGULAR NAME
- 2 PLURAL NAME
- 3 COMMAND

`\definestartstop` [^{1.}] [...,^{2.},...]

- 1 IDENTIFIER
- 2 before = COMMAND
- after = COMMAND
- style = normal bold slanted boldslanted
type cap small... COMMAND
- commands = COMMAND

`\definesynonyms` [^{1.}] [^{2.}] [^{3.}]
[^{4.}]
OPT

- 1 SINGULAR NAME
- 2 PLURAL NAME
- 3 COMMAND
- 4 COMMAND

`\definetabulate` [^{1.}] [^{2.}] [^{3.}]
OPT

- 1 IDENTIFIER
- 2 IDENTIFIER
- 3 TEXT

`\externalfigure` [^{1.}] [...,^{2.},...]
OPT

- 1 FILE
- 2 *inherits from* \setupexternalfigures

`\footnote` [^{1.}] [^{2.}]
OPT

- 1 REFERENCE
- 2 CONTENT

`\framed` [...,^{1.},...] {^{2.}}

- 1 *inherits from* \setupframed
- 2 CONTENT

`\from` [^{*}]

- * REFERENCE

`\godown` [^{*}]

- * DIMENSION

`\goto` {^{1.}} {^{2.}} [...^{3.},...]

- 1 CONTENT
- 2 CONTENT
- 3 REFERENCE

`\hairline`

`\in` {^{1.}} {^{2.}} [^{3.}]

- 1 CONTENT
- 2 CONTENT
- 3 REFERENCE

`\indenting` [...,^{*}...]

- * never none not no yes always first next
small medium big normal odd even DIMENSION



Command definitions

`\inmargin [...1...] [...2...] {...3...}`

- 1 + - low
2 REFERENCE
3 CONTENT

`\mainlanguage [...*...]`

- * n1 fr en uk de es cz ..

`\note [...*...]`

- * REFERENCE

`\page [...*...]`

- * yes makeup no preference bigpreference left
right disable last quadruple even odd blank
empty reset start stop

`\pagereference [...*...]`

- * REFERENCE

`\placefootnotes [...,*...]`

- * inherits from \setupfootnotes

`\placeformula [...1...] {...2...}`
`$$3...$$`

- 1 REFERENCE
2 CONTENT
3 DISPLAY MATH

`\placelist [...1...] [...2*,...]`

- 1 IDENTIFIER
2 inherits from \setuplist

`\placelocalfootnotes [...,*...]`

- * inherits from \setupfootnotes

`\rotate [...1*,...] {...2...}`

- 1 inherits from \setuprotate
2 CONTENT

`\scale [...1*,...] {...2...}`

- 1 sx = NUMBER
sy = NUMBER
2 CONTENT

`\section [...1*,...] {...2...}`

- 1 REFERENCE
2 CONTENT

`\setupalign [...,*...]`

- * width left right middle inner outer wide
broad height bottom line reset hanging
nohanging hyphenated nothyphenated
lesshyphenation morehyphenation new
old normal yes no flushleft flushright
flushouter flushinner center hz nohz spacing
nospacing tolerant verytolerant stretch
extremestretch lefttoright righttoleft

Command definitions

```
\setupdescriptions [...1,...]
[...=2,...]          OPT
```

1 IDENTIFIER

2 style = normal bold slanted
boldslanted type cap
small... COMMAND

color = IDENTIFIER

width = fit broad DIMENSION

distance = DIMENSION

sample = TEXT

text = TEXT

closesymbol = TEXT

closecommand = \...#1

closesymbol = TEXT

titleleft = TEXT

titleright = TEXT

titledistance = DIMENSION

titlestyle = normal bold slanted
boldslanted type cap
small... COMMAND

titlecolor = IDENTIFIER

align = inner outer left right
flushleft flushright middle
center normal no yes broad
last r2l l2r

margin = standard yes no DIMENSION

location = left right top serried
inmargin inleft inright
hanging

headstyle = normal bold slanted
boldslanted type cap
small... COMMAND

headcolor = IDENTIFIER

headcommand = COMMAND

hang = fit broad NUMBER

before = COMMAND

inbetween = COMMAND

after = COMMAND

indentnext = yes no

indenting = never none not no yes
always first next small
medium big normal odd even
DIMENSION

command = COMMAND

display = yes no

```
\setupfillinrules [...*,...]

* width = fit broad DIMENSION
  distance = DIMENSION
  before = COMMAND
  after = COMMAND
  style = normal bold slanted  
boldslanted type cap  
small... COMMAND

n = NUMBER
interlinespace = small medium big
separator = TEXT
```

```
\setupfloat [...1.] [...=2,...]

1 IDENTIFIER
2 height = DIMENSION
width = DIMENSION
maxheight = DIMENSION
maxwidth = DIMENSION
minwidth = DIMENSION
default = IDENTIFIER
pageboundaries = LIST
leftmargindistance = DIMENSION
rightmargindistance = DIMENSION
location = left middle right
inherits from \setupframed
```

```
\setupfloats [...*,...]

* location = left right middle
width = fit DIMENSION
before = COMMAND
after = COMMAND
margin = DIMENSION
spacebefore = small medium big none
spaceafter = small medium big none
sidespacebefore = small medium big none
sidespaceafter = small medium big none
indentnext = yes no
ntop = NUMBER
nbottom = NUMBER
nlines = NUMBER
default = IDENTIFIER
tolerance = 0 1 2
leftmargindistance = DIMENSION
rightmargindistance = DIMENSION
sidealign = normal line
numbering = yes nocheck
inherits from \setupframed
```

A

```
\setupenumerations [...1,...]
[...=2,...]          OPT
```

1 IDENTIFIER

2 *inherits from \setupdescriptions*

Command definitions

`\setupfooter` [¹._{OPT}.] [²..._{OPT}.²...]

- 1 *inherits from* \setupheader
- 2 *inherits from* \setupheader

`\setupfootertexts` [¹._{OPT}.] [²..._{OPT}.] [³...]

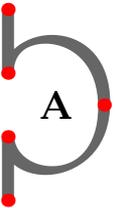
- 1 text margin edge
- 2 TEXT SECTION date MARK pagenumber
- 3 TEXT SECTION date MARK pagenumber

`\setupfootnotes` [^{*}..._{OPT}.^{*}...]

- * `conversion` = numbers characters
Characters romannumerals
Romannumerals
- `way` = bytext bycd:section
- `location` = page TEXT columns
firstcolumn lastcolumn
high none
- `rule` = on off
- `before` = COMMAND
- `after` = COMMAND
- `width` = DIMENSION
- `height` = DIMENSION
- `bodyfont` = 5pt ... 12pt small big
- `style` = normal bold slanted
boldslanted type cap
small... COMMAND
- `distance` = DIMENSION
- `columndistance` = DIMENSION
- `margindistance` = DIMENSION
- `n` = NUMBER
- `numbercommand` = \...#1
- `textcommand` = \...#1
- `split` = tolerant strict verystrict
NUMBER
- `textstyle` = normal bold slanted
boldslanted type cap
small... COMMAND
- `textcolor` = IDENTIFIER
- `interaction` = yes no
- `paragraph` = yes no
- `factor` = NUMBER
- inherits from* \setupframed

`\setupframed` [¹._{OPT}.] [²..._{OPT}.²...]

- 1 IDENTIFIER
- 2 `height` = fit broad DIMENSION
- `width` = fit broad fixed local
DIMENSION
- `autowidth` = yes no force
- `offset` = none overlay default
DIMENSION
- `location` = depth hanging high lohi
low top middle bottom
keep
- `option` = none empty
- `strut` = yes no global local
- `align` = inner outer left right
flushleft flushright
middle center normal no
yes broad last r2l l2r
- `bottom` = COMMAND
- `top` = COMMAND
- `frame` = on off none overlay
- `topframe` = on off
- `bottomframe` = on off
- `leftframe` = on off
- `rightframe` = on off
- `frameoffset` = DIMENSION
- `framedepth` = DIMENSION
- `framecorner` = round rectangular
- `frameradius` = DIMENSION
- `framecolor` = IDENTIFIER
- `background` = screen color none
foreground IDENTIFIER
- `backgroundscreen` = NUMBER
- `backgroundcolor` = IDENTIFIER
- `backgroundoffset` = frame DIMENSION
- `backgrounddepth` = DIMENSION
- `backgroundcorner` = round rectangular
- `backgroundradius` = DIMENSION
- `depth` = DIMENSION
- `corner` = round rectangular
- `radius` = DIMENSION
- `empty` = yes no
- `foregroundcolor` = IDENTIFIER
- `foregroundstyle` = normal bold slanted
boldslanted type cap
small... COMMAND
- `rulethickness` = DIMENSION



Command definitions

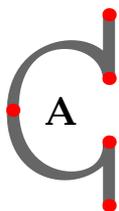
`\setupframedtexts` [\dots , $\overset{1}{\cdot}$, \dots] [\dots , $\overset{2}{\cdot}$, \dots]
OPT

1 IDENTIFIER
 2 bodyfont = 5pt ... 12pt small big
 style = normal bold slanted
 boldslanted type cap
 small... COMMAND
 left = COMMAND
 right = COMMAND
 before = COMMAND
 after = COMMAND
 inner = COMMAND
 linecorrection = on off
 depthcorrection = on off
 margin = standard yes no
 location = left right middle none
 indenting = never none not no yes
 always first next small
 medium big normal odd
 even DIMENSION

inherits from \setupframed

`\setuphead` [\dots , $\overset{1}{\cdot}$, \dots] [\dots , $\overset{2}{\cdot}$, \dots]

1 SECTION
 2 style = normal bold slanted
 boldslanted type cap
 small... COMMAND
 textstyle = normal bold slanted
 boldslanted type cap
 small... COMMAND
 numberstyle = normal bold slanted
 boldslanted type cap
 small... COMMAND
 color = IDENTIFIER
 textcolor = IDENTIFIER
 numbercolor = IDENTIFIER
 number = yes no
 ownnumber = yes no
 page = left right yes
 continue = yes no
 header = none empty high
 nomarking
 text = none empty high
 nomarking
 footer = none empty high
 nomarking
 before = COMMAND
 inbetween = COMMAND
 after = COMMAND
 alternative = normal inmargin middle
 TEXT
 hang = none broad fit line
 NUMBER
 command = \dots #1#2
 numbercommand = \dots #1
 textcommand = \dots #1
 deepnumbercommand = \dots #1
 deeptextcommand = \dots #1
 prefix = + - TEXT
 placehead = yes no empty
 incrementnumber = yes no LIST FILE
 resetnumber = yes no
 file = IDENTIFIER
 expansion = yes no command
 margintext = yes no
 interaction = list none
inherits from \setupheads



Command definitions

`\setupheader` [`.1.`] [`...,2...`]

OPT
 1 `TEXT` margin edge
 2 state = normal stop start empty high
 none nomarking IDENTIFIER
 strut = yes no
 style = normal bold slanted
 boldslanted type cap small...
 COMMAND
 leftstyle = normal bold slanted
 boldslanted type cap small...
 COMMAND
 rightstyle = normal bold slanted
 boldslanted type cap small...
 COMMAND
 leftwidth = DIMENSION
 rightwidth = DIMENSION
 before = COMMAND
 after = COMMAND

`\setupinmargin` [`.1.`] [`...,2...`]

OPT
 1 left right NUMBER
 2 location = left right both
 style = normal bold slanted boldslanted
 type cap small... COMMAND
 before = COMMAND
 after = COMMAND
 align = inner outer left right
 flushleft flushright middle
 center normal no yes broad last
 r2l l2r
 line = NUMBER
 distance = DIMENSION
 separator = TEXT
 width = DIMENSION
 distance = DIMENSION
 stack = yes no
inherits from \setupframed

`\setupheadertexts` [`.1.`] [`.2.`] [`.3.`]

OPT
 1 `text` margin edge
 2 TEXT SECTION date MARK pagenumber
 3 TEXT SECTION date MARK pagenumber

`\setupinteraction` [`...,*...`]

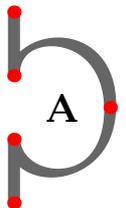
* state = start stop
 menu = on off
 page = yes no
 click = yes no
 split = yes no
 display = new
 openaction = REFERENCE
 closeaction = REFERENCE
 openpageaction = REFERENCE
 closepageaction = REFERENCE
 calculate = IDENTIFIER
 strut = yes no
 width = DIMENSION
 height = DIMENSION
 depth = DIMENSION
 style = normal bold slanted
 boldslanted type cap
 small... COMMAND
 color = IDENTIFIER
 contrastcolor = IDENTIFIER
 symbolset = IDENTIFIER
 title = TEXT
 subtitle = TEXT
 author = TEXT
 date = TEXT
 keyword = TEXT
 fieldlayer = auto IDENTIFIER

`\setupheads` [`...,*...`]

* sectionnumber = yes NUMBER no
 alternative = normal margin middle TEXT
 paragraph
 separator = TEXT
 stopper = TEXT
 align = inner outer left right
 flushleft flushright middle
 center normal no yes broad
 last r2l l2r
 alightitle = yes float no
 tolerance = verystRICT strict tolerant
 verytolerant stretch
 indentnext = yes no
 command = `\...#1#2`
 margin = DIMENSION

`\setupindenting` [`...,*...`]

* never none not no yes always first next
 small medium big normal odd even DIMENSION



Command definitions

`\setupinteractionmenu [...1;...]
[...2...]`

1 IDENTIFIER
2 before = COMMAND
after = COMMAND
inbetween = COMMAND
left = COMMAND
right = COMMAND
middle = COMMAND
state = start stop none local
style = normal bold slanted
boldslanted type cap
small... COMMAND

color = IDENTIFIER
contrastcolor = IDENTIFIER
distance = overlay DIMENSION
samepage = yes empty no none
unknownreference = yes empty no none
leftoffset = DIMENSION
rightoffset = DIMENSION
topoffset = DIMENSION
bottomoffset = DIMENSION
position = yes no
inherits from \setupframed

`\setuplanguage [...1] [...2...]`

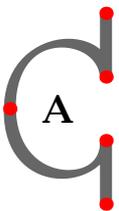
1 nl fr en uk de es cz ...
2 *inherits from \installlanguage*

`\setuplayout [...,*...]`

* width = DIMENSION fit middle
height = DIMENSION fit middle
backspace = DIMENSION
topspace = DIMENSION
margin = DIMENSION
leftmargin = DIMENSION
rightmargin = DIMENSION
header = DIMENSION
footer = DIMENSION
top = DIMENSION
bottom = DIMENSION
leftedge = DIMENSION
rightedge = DIMENSION
headerdistance = DIMENSION
footerdistance = DIMENSION
topdistance = DIMENSION
bottomdistance = DIMENSION
leftmargindistance = DIMENSION
rightmargindistance = DIMENSION
leftedgedistance = DIMENSION
rightedgedistance = DIMENSION
horoffset = DIMENSION
veroffset = DIMENSION
style = normal bold slanted
boldslanted type cap
small... COMMAND

color = IDENTIFIER
marking = on off color screen
TEXT page
location = left middle
right bottom
top singlesided
doublesided

scale = DIMENSION
nx = NUMBER
ny = NUMBER
dx = DIMENSION
dy = DIMENSION
lines = NUMBER
columns = NUMBER
columndistance = DIMENSION
grid = yes no
bottomspace = DIMENSION
cutspace = DIMENSION
textdistance = DIMENSION
textwidth = NUMBER
textmargin = DIMENSION
clipoffset = DIMENSION
page = IDENTIFIER
paper = IDENTIFIER



Command definitions

`\setuplist [...,1...] [...,2,...]`

1 IDENTIFIER
 2 state = start stop
 alternative = a b c ... none command
 coupling = on off
 criterium = SECTION local previous
 current here intro all
 extras = LIST
 style = normal bold slanted
 boldslanted type cap
 small... COMMAND
 numberstyle = normal bold slanted
 boldslanted type cap
 small... COMMAND
 textstyle = normal bold slanted
 boldslanted type cap
 small... COMMAND
 pagestyle = normal bold slanted
 boldslanted type cap
 small... COMMAND
 color = IDENTIFIER
 command = \...#1#2#3
 numbercommand = \...#1
 textcommand = \...#1
 pagecommand = \...#1
 interaction = sectionnumber text
 pagenumber all none
 before = COMMAND
 after = COMMAND
 inbetween = COMMAND
 left = TEXT
 right = TEXT
 label = yes no
 prefix = yes no none
 pagenumber = yes no
 headnumber = yes no
 cd:sectionnumber = yes no
 aligntitle = yes no
 margin = DIMENSION
 width = DIMENSION fit
 height = DIMENSION fit broad
 depth = DIMENSION fit broad
 distance = DIMENSION
 separator = TEXT
 stopper = TEXT
 symbol = none 1 2 3 ...
 expansion = yes no command
 maxwidth = DIMENSION
 order = command all normal
inherits from \setupframed

`\setupmakeup [...,1...] [...,2,...]`

1 IDENTIFIER
 2 width = DIMENSION
 height = DIMENSION
 page = left yes right
 commands = COMMAND
 doublesided = yes no empty
 headerstate = normal stop start empty none
 nomarking
 footerstate = normal stop start empty none
 nomarking
 textstate = normal stop start empty none
 nomarking
 topstate = stop start
 bottomstate = stop start
 pagestate = stop start
 color = IDENTIFIER

`\setupmarginblocks [...,*...]`

* location = inmargin left middle right
 style = normal bold slanted boldslanted
 type cap small... COMMAND
 width = DIMENSION
 align = inner outer left right
 flushleft flushright middle
 center normal no yes broad last
 r2l l2r
 top = COMMAND
 inbetween = COMMAND
 bottom = COMMAND
 left = COMMAND
 right = COMMAND
 before = COMMAND
 after = COMMAND



Command definitions

`\setuppagenumbering [...,.*,...]`

* alternative = singlesided doublesided
 location = header footer left right
 middle margin marginedge
 inleft inright
 conversion = numbers characters
 Characters romannumerals
 Romannumerals
 style = normal bold slanted
 boldslanted type cap
 small... COMMAND
 left = TEXT
 right = TEXT
 way = bytext bycd:section
 text = TEXT
 numberseparator = TEXT
 textseparator = TEXT
 cd:sectionnumber = yes no
 separator = TEXT
 strut = yes no
 state = start stop
 width = DIMENSION
 command = \...#1
 page = last yes no

`\setupparagraphs [...,.*,...]`
`[...,.3.,...]` OPT

1 IDENTIFIER
 2 NUMBER each
 3 style = normal bold slanted boldslanted
 type cap small... COMMAND
 width = DIMENSION
 height = DIMENSION
 align = inner outer left right
 flushleft flushright middle
 center normal no yes broad last
 r2l l2r
 tolerance = verystRICT strict tolerant
 verytolerant stretch
 distance = DIMENSION
 before = COMMAND
 after = COMMAND
 inner = COMMAND
 command = COMMAND
 rule = on off

`\setupregister [...,.*,...]`
`[...,.3.,...]` OPT

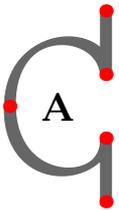
1 SINGULAR NAME
 2 IDENTIFIER
 3 n = NUMBER
 balance = yes no
 align = inner outer left right
 flushleft flushright
 middle center normal no
 yes broad last r2l l2r
 style = normal bold slanted
 boldslanted type cap
 small... COMMAND
 pagestyle = normal bold slanted
 boldslanted type cap
 small... COMMAND
 textstyle = normal bold slanted
 boldslanted type cap
 small... COMMAND
 indicator = yes no
 coupling = yes no
 cd:sectionnumber = yes no
 criterium = SECTION local all
 distance = DIMENSION
 symbol = 1 2 ... n a ... none
 interaction = pagenumber TEXT
 expansion = yes no command
 referencing = on off
 command = \...#1
 location = left middle right
 maxwidth = DIMENSION
 unknownreference = empty none
 alternative = a b A B
 prefix = both first none
 compress = no yes
 deeptextcommand = \...#1

`\setuprotate [...,.*,...]`

* rotation = NUMBER
 location = normal high fit broad depth
inherits from \setupframed

`\setupsectionblock [...,.*,...]`

1 IDENTIFIER
 2 number = yes no
 page = yes right
 before = COMMAND
 after = COMMAND



Command definitions

`\setuptabulate` [₁...] [₂...] [_{OPT}...]

1 IDENTIFIER
 2 unit = DIMENSION
 indenting = never none not no yes
 always first next small
 medium big normal odd even
 DIMENSION
 before = COMMAND
 after = COMMAND
 inner = COMMAND
 EQ = TEXT
 rulecolor = IDENTIFIER
 align = inner outer left right
 flushleft flushright middle
 center normal no yes broad
 last r2l l2r
 rulethickness = DIMENSION
 distance = blank grid depth DIMENSION
 small medium big none
 bodyfont = 5pt ... 12pt small big
 rule = normal line
 split = yes no

`\setupthinrules` [_{*}...]

* interlinespace = small medium big
 n = NUMBER
 before = COMMAND
 inbetween = COMMAND
 after = COMMAND
 color = IDENTIFIER
 backgroundcolor = IDENTIFIER
 height = DIMENSION max
 depth = DIMENSION max
 alternative = a b c d
 rulethickness = DIMENSION
 color = IDENTIFIER
 background = color
 backgroundcolor = IDENTIFIER

`\setuptype` [_{*}...] [_{OPT}...]

* space = on off
 option = slanted normal none
 style = normal bold slanted boldslanted
 type cap small... COMMAND
 color = IDENTIFIER
 left = COMMAND
 right = COMMAND
 command = COMMAND

`\setuptyping` [₁...] [₂...] [_{OPT}...]

1 file typing IDENTIFIER
 2 space = on off
 page = yes no
 option = slanted normal commands color
 none
 text = yes no
 icommand = COMMAND
 vcommand = COMMAND
 ccommand = COMMAND
 before = COMMAND
 after = COMMAND
 margin = DIMENSION standard yes no
 evenmargin = DIMENSION
 oddmargin = DIMENSION
 blank = DIMENSION small medium big
 standard halfline line
 escape = TEXT
 space = on off
 tab = NUMBER yes no
 page = yes no
 indentnext = yes no
 style = normal bold slanted
 boldslanted type cap small...
 COMMAND
 color = IDENTIFIER
 palet = IDENTIFIER
 range = TEXT
 lines = yes no hyphenated
 empty = yes all no
 numbering = line file no
 bodyfont = 5pt ... 12pt small big
 strip = no auto NUMBER

`\setupwhitespace` [_{*}...]
 OPT

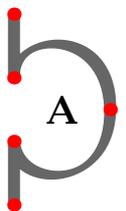
* none small medium big line fixed fix
 DIMENSION

`\startalignment` [_{*}...] ...
`\stopalignment` OPT

* *inherits from* \setupalign

`\startbuffer` [_{*}...] ... \stopbuffer
 OPT

* IDENTIFIER



Command definitions

`\startcolumns [...,*...]` ...
`\stopcolumns`

* inherits from `\setupcolumns`

`\startcombination [...]` ...
`\stopcombination`

* N*M

`\startcomment [.1.] [...,.2.,...]` ...
`\stopcomment`

1 IDENTIFIER
2 inherits from `\setupcomment`

`\starthiding` ... `\stophiding`

`\startinteractionmenu [...]` ...
`\stopinteractionmenu`

* IDENTIFIER

`\startlinecorrection` ...
`\stoplinecorrection`

`\startlines` ... `\stoplines`

`\startlocalfootnotes` ...
`\stoplocalfootnotes`

`\startpacked [...]` ... `\stoppacked`

* blank

`\startquotation [...,*...]` ...
`\stopquotation`

* left middle right

`\starttable [...]` ... `\stoptable`

* TEXT IDENTIFIER

`\startunpacked` ... `\stopunpacked`

`\subject [...1;...]` {`...2.`}

1 REFERENCE
2 CONTENT

`\subsection [...1;...]` {`...2.`}

1 REFERENCE
2 CONTENT

`\subsubject [...1;...]` {`...2.`}

1 REFERENCE
2 CONTENT

`\switchtobodyfont [...,*...]`

* IDENTIFIER serif regular roman sans support
sansserif mono type teletype handwritten
calligraphic 5pt ... 12pt small big

Command definitions

`\textreference [..1.] {..2.}`

- 1 REFERENCE
- 2 CONTENT

`\thinrule`

`\thinrules [..*.]`
OPT

* *inherits from* \setupthinrules

`\title [..1,..2.] {..3.}`
OPT

- 1 REFERENCE
- 2 CONTENT

`\type {..*.}`

* CONTENT

`\useexternaldocument [..1.] [..2.]
[..3.]`

- 1 IDENTIFIER
- 2 file
- 3 TEXT

`\usemodule [..*,...]`

* IDENTIFIER

`\useURL [..1.] [..2.] [..3.] [..4.]`

- 1 IDENTIFIER
- 2 TEXT
- 3 file
- 4 TEXT

`\writebetweenlist [..1.] [..2,...]
{..3.}`

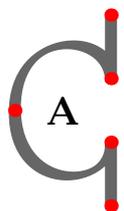
- 1 SECTION IDENTIFIER
- 2 location = here none
- 3 CONTENT

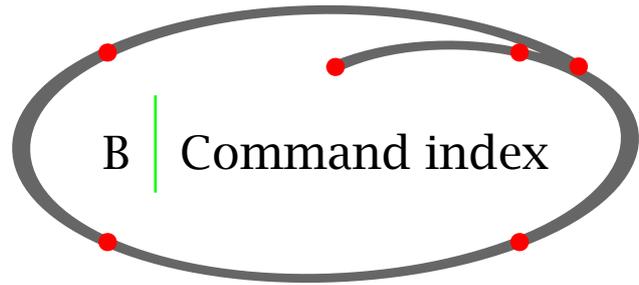
`\writetolist [..1.] [..2,...]
{..3.}`

- 1 SECTION IDENTIFIER
- 2 location = here none
- 3 CONTENT
- 4 CONTENT

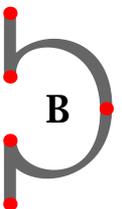


Command definitions





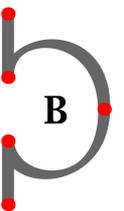
<code>\abbreviation</code> 65	<code>\definetabulate</code> 41
<code>\adaptlayout</code> 83	<code>\definetextbackground</code> 90
<code>\at</code> 68, 74	<code>\em</code> 82
<code>\bf</code> 81	<code>\en</code> 110
<code>\blank</code> 92	<code>\eTABLE</code> 34
<code>\bTABLE</code> 34	<code>\eTD</code> 34
<code>\bTD</code> 34	<code>\eTR</code> 34
<code>\bTR</code> 34	<code>\externalfigure</code> 29
<code>\cap</code> 81	<code>\FLOWchart</code> 111
<code>\chapter</code> 11	<code>\FR</code> 34
<code>\chemical</code> 26	<code>\fixedspaces</code> 109
<code>\color</code> 71	<code>\footnote</code> 46
<code>\column</code> 43	<code>\framed</code> 52
<code>\completecontent</code> 61	<code>\from</code> 74
<code>\completeindex</code> 64	<code>\getbuffer</code> 103
<code>\completelistofabbreviations</code> 65	<code>\getvariable</code> 101
<code>\completelistofsorts</code> 66	<code>\goto</code> 74
<code>\completeregister</code> 64	<code>\HL</code> 34
<code>\crlf</code> 109	<code>\hairline</code> 104
<code>\currentdate</code> 107	<code>\head</code> 14
<code>\DC</code> 34	<code>\high</code> 106
<code>\DL</code> 34	<code>\hskip</code> 94
<code>\DR</code> 34	<code>\in</code> 68, 74
<code>\definebodyfont</code> 81	<code>\indenting</code> 94
<code>\definecolor</code> 71	<code>\index</code> 64
<code>\definecombinedlist</code> 61	<code>\inframed</code> 52
<code>\definedescription</code> 49	<code>\infull</code> 65
<code>\defineenumeration</code> 51	<code>\inleft</code> 56
<code>\definefloat</code> 101	<code>\inmargin</code> 56
<code>\definelist</code> 61	<code>\input</code> 113
<code>\definemakeup</code> 97	<code>\inright</code> 56
<code>\defineregister</code> 64	<code>\item</code> 14
<code>\definesorting</code> 66	<code>\LOW</code> 34
<code>\definesymbol</code> 14	<code>\LR</code> 34
<code>\definesynonyms</code> 65	<code>\language</code> 110

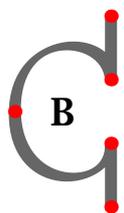


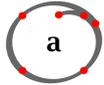
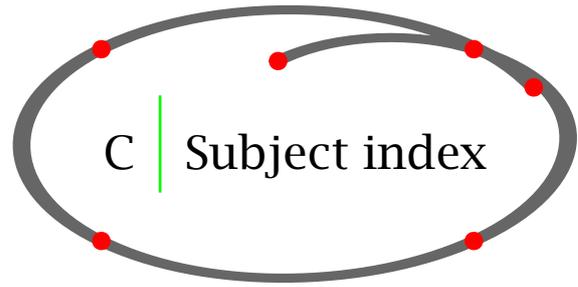
`\leftlines` 72
`\loadabbreviations` 65
`\lohi` 106
`\low` 106
`\MR` 34
`\mainlanguage` 110
`\margintext` 56
`\margintitle` 56
`\midaligned` 72
`\NC` 34, 41
`\NR` 34, 41
`\nI` 110
`\noheaderandfooterlines` 60
`\noindenting` 94
`\nowhitespace` 92
`\overstrikes` 104
`\page` 57
`\pagereference` 68
`\par` 91
`\paragraph` 11, 91
`\periods` 104
`\placecontent` 61
`\placefigure` 29
`\placeformula` 24
`\placeindex` 64
`\placeintermezzo` 101
`\placelistofabbreviations` 65
`\placelistofsorts` 66
`\placepublications` 29
`\placeregister` 64
`\placetable` 34
`\quote` 48
`\rightaligned` 72
`\rm` 81
`\rotate` 107
`\SR` 34
`\savebuffer` 103
`\scale` 108
`\setup` 100
`\setup tolerance` 72
`\setupalign` 72
`\setupbackgrounds` 89
`\setupbibtex` 29
`\setupblank` 92
`\setupbodyfont` 78
`\setupbuffer` 103
`\setupcaptions` 29, 34, 101
`\setupcolors` 71
`\setupcolumns` 43
`\setupcombinedlist` 61
`\setupdescriptions` 49
`\setupenumerations` 51
`\setupfigures` 29
`\setupfloat` 101
`\setupfloats` 29, 34, 101
`\setupfooter` 60
`\setupfootertexts` 60
`\setupfootnotes` 46
`\setupformulas` 24
`\setupframed` 52
`\setupframedtext` 54
`\setuphead` 11
`\setupheader` 60
`\setupheadertexts` 60
`\setupheads` 11
`\setupindenting` 94
`\setupinteraction` 74
`\setupitemize` 14
`\setuplayout` 83
`\setuplist` 61
`\setupmakeup` 97
`\setuppagenumbering` 57
`\setuppublications` 29
`\setupregister` 64
`\setupscale` 108
`\setupsorting` 66
`\setupsynonyms` 65
`\setupTABLE` 34
`\setuptables` 34
`\setuptabulate` 41
`\setuptextbackground` 90
`\setupthinrules` 104
`\setuptype` 82
`\setuptyping` 82
`\setupuserpagenumber` 57
`\setupwhitespace` 92
`\setupxtable` 34
`\setvariables` 101

Command index

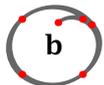
`\showframe` 83
`\showlayout` 83
`\showmakeup` 83
`\showsetups` 83
`\sl` 81
`\sort` 66
`\space` 109
`\ss` 81
`\startbuffer` 103
`\startcolumns` 43
`\startcombination` 29, 34
`\startcomment` 113
`\startfiguretext` 29
`\startformula` 24
`\startframedtext` 54
`\starthiding` 113
`\startitemize` 14
`\startlinecorrection` 92
`\startlines` 109
`\startlocal` 83
`\startpacked` 92
`\startstandardmakeup` 97
`\starttable` 34
`\starttabulate` 41
`\starttextbackground` 90
`\starttextrule` 104
`\starttyping` 82
`\startunpacked` 92
`\startxcell` 34
`\startxrow` 34
`\startxtable` 34
`\stopxtable` 34
`\subject` 11
`\subparagraph` 11
`\subsubject` 11
`\switchtobodyfont` 78
`\THREE` 34
`\TWO` 34
`\tfa` 81
`\tfb` 81
`\tfc` 81
`\tfd` 81
`\thinrule` 104
`\thinrules` 104
`\title` 11
`\tt` 81
`\type` 82
`\typebuffer` 103
`\underbar` 104
`\unit` 27
`\useexternaldocument` 74
`\usemodule` 114
`\VL` 34
`\vskip` 94
`\whitespace` 92
`\writebetweenlist` 61
`\writetolist` 61



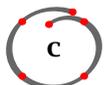




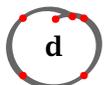
AMS 78
accents 83
alignment 72
auxilliary files 155



BIB_TE_X 29
background
 page areas 89
 text 90
bibliography 29

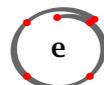


Computer Modern Roman 78
carriage return 109
chart 111
chemical equitation 26
chemistry 26
citation 48
color 71
columns 41, 43
columns in itemize 14
comment 112



date 107

definition 49
display mode 19



emphasized 82
errors 149
extreme tables 34

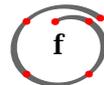
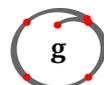
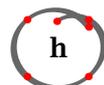


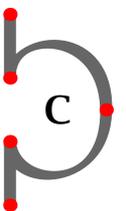
figure 29
floating blocks 29, 34, 101
font
 size 78
 style 78
fontsize 81
footnote 46
foreign characters 83
formula 24

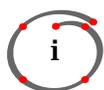


graphical features 116

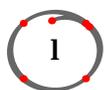


headers 11
hiding text 113
hyphenation 110

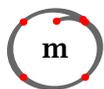




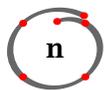
indentation 94
input file 5, 117
 processing 6
input other TeX-files 113
inter paragraph spacing 92
interaction
 external 74
 internal 74
interactive mode 74
itemize 14



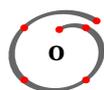
Lucida Bright 78
label 68
language 110
layout 10, 83
lines 104
list 61



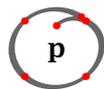
margin text 56
math 18
math mode 19
mathml 114
METAPOST 116
module 114



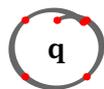
natural tables 34
non-breakable space 109
note 113
numbered definition 51



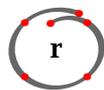
openmath 114
outline
 paragraph 54
 text 52
overlay 98



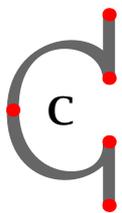
page areas 89
page breaking 57
page design 83
page footer 60
page header 60
page numbering 57
paragraph 91
 indentation 94
pdf-file 6
% in input file 112
picture *see figure*
postponing a block 101
preamble 117
presentation 115
problems 149
processing TeX file 153



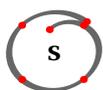
quote 48



ACROBAT READER 73
referring 68
register 64
rotating 107



Subject index



scaling 108
set up 10
setups 100
SI-unit 27
small caps 81
sorted lists 66
space 109
special characters 7
storing text 103
subscript 106
superscript 106
support 143
symbols in math mode 148
synonyms 65

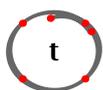
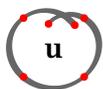
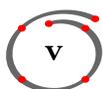


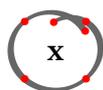
table of contents 61
tables 34
 running text 41
tabulation 41
text mode 19
tilde 109
titlepage 97
tuc-file 155
type 82
context (command) 153
mtxrun (command) 153



units 27



variables 101
verbatim 82

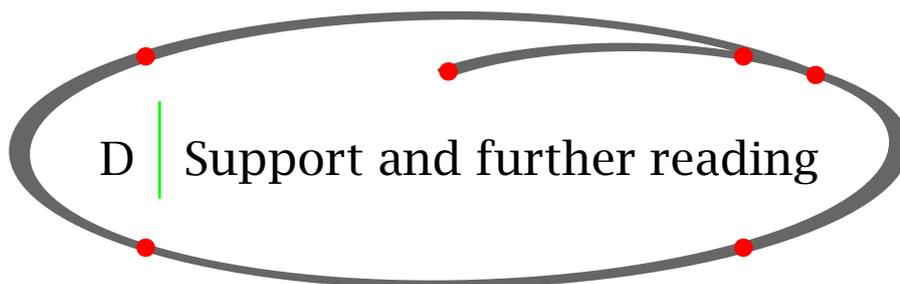


xml 114



Subject index





D.1

For your Questions and Answers you can subscribe to the `CONTEX`T mailing list.

Visit the Pragma ADE website for extensive information about `CONTEX`T.

Goto the ConTeXt Garden to find all kinds of practical information on how to use `CONTEX`T.

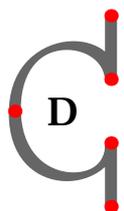
D.2 Manuals

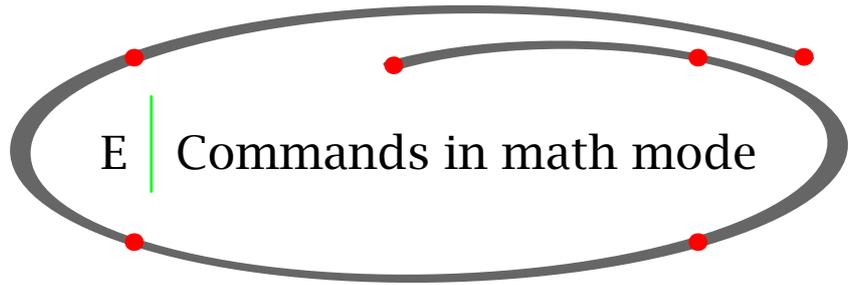
Chemical Formulas in `CONTEX`TColor SeparationColumns`CONTEX`T, the manualDealing with XMLExtreme TablesFiguresFonts in `CONTEX`Tluatools, `mtxrun`, `contextMETAFUN` manual-Natural TablesPPCH`TEX` ManualQuick Reference (dutch)Quick Reference (english)SCITE in `CONTEX`TUnitsWidgets

D.3 Magazines

`CONTEX`T Magazine 1103Project structure







E.1 Greek characters

α	<code>\alpha</code>	ι	<code>\iota</code>	ϱ	<code>\varrho</code>
β	<code>\beta</code>	κ	<code>\kappa</code>	Σ	<code>\sigma</code>
Γ	<code>\gamma</code>	Λ	<code>\lambda</code>	ζ	<code>\varsigma</code>
Δ	<code>\delta</code>	μ	<code>\mu</code>	τ	<code>\tau</code>
ϵ	<code>\epsilon</code>	ν	<code>\nu</code>	υ	<code>\upsilon</code>
ε	<code>\varepsilon</code>	Ξ	<code>\xi</code>	Φ	<code>\phi</code>
ζ	<code>\zeta</code>	o	<code>o</code>	φ	<code>\varphi</code>
η	<code>\eta</code>	Π	<code>\pi</code>	χ	<code>\chi</code>
Θ	<code>\theta</code>	ϖ	<code>\varpi</code>	Ψ	<code>\psi</code>
ϑ	<code>\vartheta</code>	ρ	<code>\rho</code>	Ω	<code>\omega</code>

E.2 Special symbols

\aleph	<code>\aleph</code>	$'$	<code>\prime</code>	\forall	<code>\forall</code>
\hbar	<code>\hbar</code>	\emptyset	<code>\emptyset</code>	\exists	<code>\exists</code>
i	<code>\imath</code>	∇	<code>\nabla</code>	\neg	<code>\neg</code>
J	<code>\jmath</code>	$\sqrt{\quad}$	<code>\surd</code>	\flat	<code>\flat</code>
ℓ	<code>\ell</code>	\top	<code>\top</code>	\natural	<code>\natural</code>
\wp	<code>\wp</code>	\perp	<code>\perp</code>	\sharp	<code>\sharp</code>
\Re	<code>\Re</code>	$\ $	<code>\Vert</code>	\clubsuit	<code>\clubsuit</code>
\Im	<code>\Im</code>	\angle	<code>\angle</code>	\diamondsuit	<code>\diamondsuit</code>
∂	<code>\partial</code>	\triangle	<code>\triangle</code>	\heartsuit	<code>\heartsuit</code>
∞	<code>\infty</code>	\backslash	<code>\backslash</code>	\spadesuit	<code>\spadesuit</code>

E.3 Operators in addition to +

\pm	<code>\pm</code>	\cap	<code>\cap</code>	\vee	<code>\vee</code>
\mp	<code>\mp</code>	\cup	<code>\cup</code>	\wedge	<code>\wedge</code>
\setminus	<code>\setminus</code>	\oplus	<code>\oplus</code>	\otimes	<code>\otimes</code>
\cdot	<code>\cdot</code>	\sqcap	<code>\sqcap</code>	\ominus	<code>\ominus</code>
\times	<code>\times</code>	\sqcup	<code>\sqcup</code>	\otimes	<code>\otimes</code>
\ast	<code>\ast</code>	\triangleleft	<code>\triangleleft</code>	\oslash	<code>\oslash</code>
\star	<code>\star</code>	\triangleright	<code>\triangleright</code>	\odot	<code>\odot</code>
\diamond	<code>\diamond</code>	\wr	<code>\wr</code>	\dagger	<code>\dagger</code>



◦ <code>\circ</code>	◯ <code>\bigcirc</code>	‡ <code>\ddagger</code>
• <code>\bullet</code>	△ <code>\bigtriangleup</code>	∏ <code>\amalg</code>
÷ <code>\div</code>	▽ <code>\bigtriangledown</code>	

E.4 Operators

∑ <code>\sum</code>	∏ <code>\prod</code>	∐ <code>\coprod</code>
∫ <code>\int</code>	∫̸ <code>\oint</code>	∩ <code>\bigcap</code>
∪ <code>\bigcup</code>	∪̸ <code>\bigsqcup</code>	∨ <code>\bigvee</code>
∧ <code>\bigwedge</code>	⊙ <code>\bigodot</code>	⊗ <code>\bigotimes</code>
⊕ <code>\bigoplus</code>	⊕̸ <code>\biguplus</code>	

E.5 Relation in addition to >

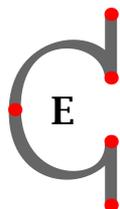
≤ <code>\leq</code>	≥ <code>\geq</code>	≡ <code>\equiv</code>
< <code>\prec</code>	> <code>\succ</code>	~ <code>\sim</code>
≲ <code>\preceq</code>	≳ <code>\succeq</code>	≈ <code>\simeq</code>
≪ <code>\ll</code>	≫ <code>\gg</code>	∝ <code>\asymp</code>
⊂ <code>\subset</code>	⊃ <code>\supset</code>	≈ <code>\approx</code>
⊆ <code>\subseteq</code>	⊇ <code>\supseteq</code>	≅ <code>\cong</code>
⊈ <code>\sqsubseteq</code>	⊉ <code>\sqsupseteq</code>	⊗ <code>\bowtie</code>
∈ <code>\in</code>	∋ <code>\ni</code>	∞ <code>\propto</code>
⊢ <code>\vdash</code>	⊣ <code>\dashv</code>	⊢ <code>\models</code>
⋈ <code>\smile</code>	<code>\mid</code>	≐ <code>\doteq</code>
⋈ <code>\frown</code>	∥ <code>\parallel</code>	⊥ <code>\perp</code>

E.6 Negated relations

≠ <code>\not<</code>	≠ <code>\not></code>	≠ <code>\not=</code>
≠ <code>\not\leq</code>	≠ <code>\not\geq</code>	≠ <code>\not\equiv</code>
≠ <code>\not\prec</code>	≠ <code>\not\succ</code>	≠ <code>\not\sim</code>
≠ <code>\not\preceq</code>	≠ <code>\not\succeq</code>	≠ <code>\not\simeq</code>
≠ <code>\not\subset</code>	≠ <code>\not\supset</code>	≠ <code>\not\approx</code>
≠ <code>\not\subseteq</code>	≠ <code>\not\supseteq</code>	≠ <code>\not\cong</code>
≠ <code>\not\sqsubseteq</code>	≠ <code>\not\sqsupseteq</code>	≠ <code>\not\asymp</code>

E.7 Some arrows

← <code>\leftarrow</code>	← <code>\longleftarrow</code>	↑ <code>\uparrow</code>
⇐ <code>\Leftarrow</code>	⇐ <code>\Longleftarrow</code>	⇑ <code>\Uparrow</code>
→ <code>\rightarrow</code>	→ <code>\longrightarrow</code>	↓ <code>\downarrow</code>
⇒ <code>\Rightarrow</code>	⇒ <code>\Longrightarrow</code>	⇓ <code>\Downarrow</code>
↔ <code>\leftrightarrow</code>	↔ <code>\longleftrightarrow</code>	↕ <code>\updownarrow</code>
⇔ <code>\Leftrightarrow</code>	⇔ <code>\Longleftrightarrow</code>	⇕ <code>\Updownarrow</code>



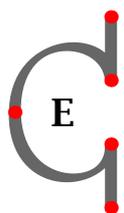
Commands in math mode

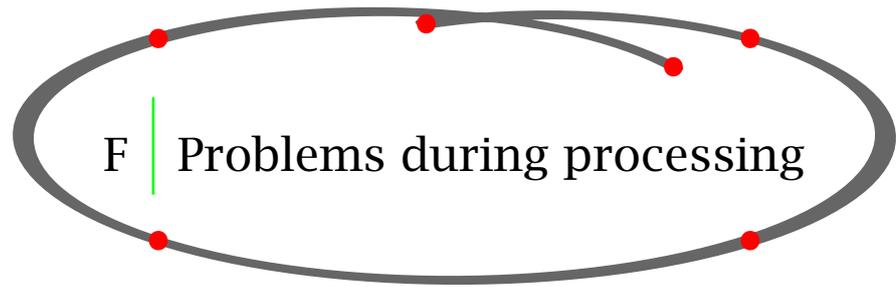
\mapsto	<code>\mapsto</code>	\longmapsto	<code>\longmapsto</code>	\nearrow	<code>\nearrow</code>
\searrow	<code>\searrow</code>	\swarrow	<code>\swarrow</code>	\nwarrow	<code>\nwarrow</code>
\hookleftarrow	<code>\hookleftarrow</code>	\hookrightarrow	<code>\hookrightarrow</code>		

E.8 Alternative commands

\neq	<code>\neq</code>	$\{ \}$	<code>{ }</code>	\wedge	<code>\land</code>	\rightarrow	<code>\to</code>	$ $	<code>\vert</code>
\leq	<code>\leq</code>	$\} \}$	<code>} }</code>	\vee	<code>\lor</code>	\leftarrow	<code>\gets</code>	\parallel	<code>\Vert</code>
\geq	<code>\geq</code>	\exists	<code>\owns</code>	\neg	<code>\lnot</code>				







If processing is not successful—for example because you typed `\stptext` instead of `\stoptext`—`CONTEXT` produces a `?` on your screen and tells you it has just processed an error. It will give you some basic information on the type of error and the line number where the error becomes effective.

At the instant of `?` you can type:

- H for help information on your error
- I for inserting the correct `CONTEXT` command
- Q for quitting and entering batch mode
- X for exiting the running mode
- ENTER for ignoring the error

Most of the time you will type `ENTER` and processing will continue. Then you can edit the input file and fix the error.

Some errors will produce a `*` on your screen and processing will stop. This error is due to a fatal error in your input file. You can't ignore this error and the only option you have is to type `\stop` or `Ctrl Z`. The program will be halted and you can fix the error in your text editor.

A well known error is:

```
! I can't write on file 'myfile.pdf'.  
Please type another filename for output:
```

This error is due to the fact that the file `myfile.pdf` is still open in `ACROBAT READER`.

The best way to proceed is:

- close the file in `ACROBAT READER`
- type `ENTER` at the console

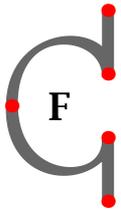
Sometimes the error messages are very obscure. Finding the location of the error in an extensive document can then be a tedious job. You could try to isolate the error:

- open the file in your text editor
- save a copy of your file (to be on the safe side)
- isolate the error
 1. place a `\stoptext` command higher up in your text
 2. process the file
 3. repeat step 1 and 2 until the file processes correctly
- study the content that produces the error

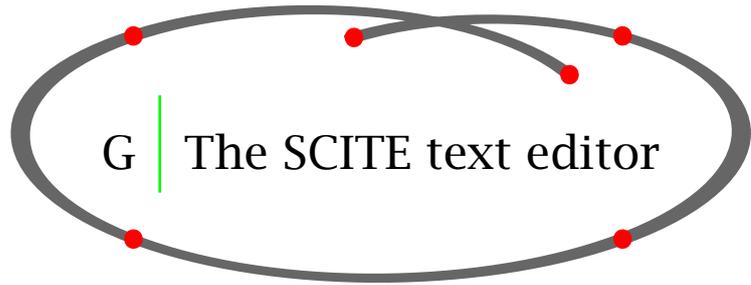


Problems during processing

- fix the error
- place the `\stoptext` command after the corrected error
- process your file
- etc.



The SCITE text editor



The developers of $\text{CON}\text{T}_\text{E}\text{X}\text{T}$ have always been able to process their $\text{T}_\text{E}\text{X}$ files from a text editor. In that way $\text{CON}\text{T}_\text{E}\text{X}\text{T}$ became an effective authoring tool.

At this moment the text editors SCITE and $\text{T}_\text{E}\text{X}\text{WORKS}$ are more or less part of the $\text{CON}\text{T}_\text{E}\text{X}\text{T}$ distribution.

Please refer to the $\text{CON}\text{T}_\text{E}\text{X}\text{T}$ WIKI and learn how to install SCITE.

SCITE supports the:

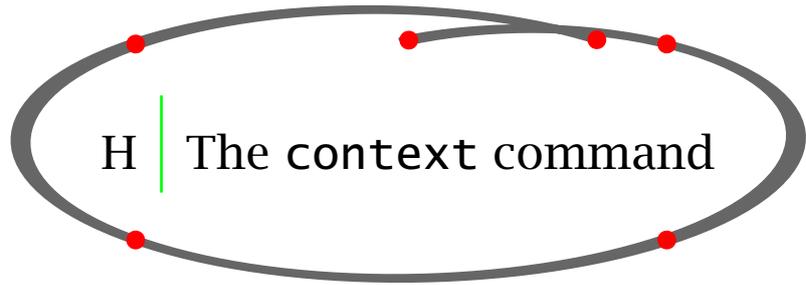
- processing $\text{T}_\text{E}\text{X}$ of files
- colored display of commands (lexing)
- syntax checking of $\text{T}_\text{E}\text{X}$, XML and LUA files
- spell checking of your text

The $\text{CON}\text{T}_\text{E}\text{X}\text{T}$ specific support of SCITE is described in the manual SCITE in $\text{CON}\text{T}_\text{E}\text{X}\text{T}$.





The context command



You can process a $\text{T}_{\text{E}}\text{X}$ file or run $\text{CON}_{\text{T}}\text{E}_{\text{X}}\text{T}$ with the command `context` that you can type at your console:

```
context myfile
```

$\text{CON}_{\text{T}}\text{E}_{\text{X}}\text{T}$ will make multiple runs to get the layout, references, lists and pag numbering straight. You can see those runs echoed on your screen and listed in the `myfile.log` file.

You can add parameters to give the command `context` additional tasks while processing the file. If you want start up ACROBAT READER automatically you can type:

```
context --autopdf myfile
```

A full overview of the parameters is given when you type:

```
context --help
```

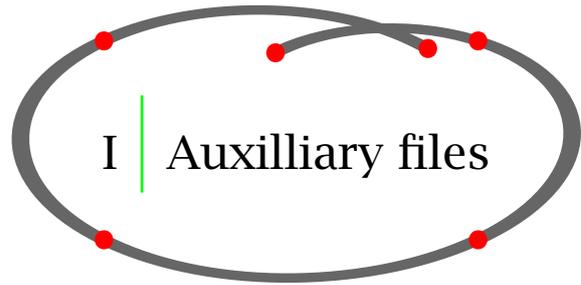
Please refer to the manual `luatools`, `mtxrun`, `context` for more information on running $\text{CON}_{\text{T}}\text{E}_{\text{X}}\text{T}$.



The context command



Auxilliary files

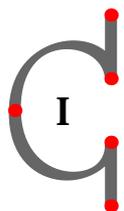


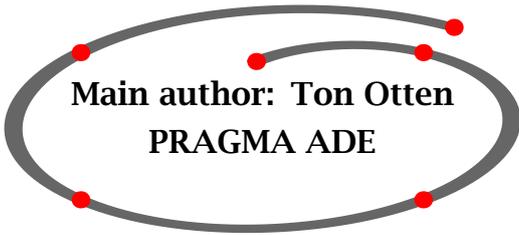
CON_TE_XT will produce a number of auxilliary files during processing. If your input file is called `myfile.tex` the following files may appear on your working directory.

CON _T E _X T MkII	CON _T E _X T MkIV	Meaning
<code>myfile.tex</code>	<code>myfile.tex</code>	your text file
<code>myfile.log</code>	<code>myfile.log</code>	log information
<code>myfile.tuo</code>	<code>myfile.tuc</code>	output information
<code>myfile.tui</code>		input information
<code>myfile.tmp</code>		temporary information
<code>mpgraph.mp</code>		METAPOST information
<code>myfile.pdf</code>	<code>myfile.pdf</code>	result file

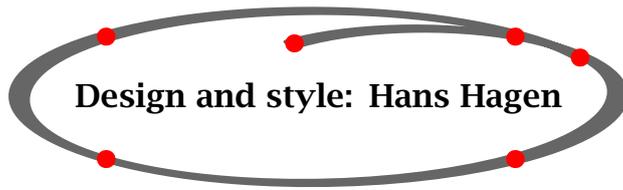
The `myfile.tuc` file contains information about registers, lists and references which will be used when necessary. The `myfile.log` can be viewed in case there are problems during processing.



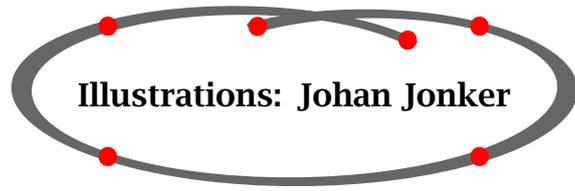




**Main author: Ton Otten
PRAGMA ADE**



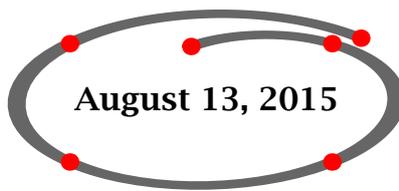
Design and style: Hans Hagen



Illustrations: Johan Jonker



**PRAGMA ADE
Ridderstraat 27
8061GH Hasselt NL
www.pragma-ade.com**



August 13, 2015

This manual describes some features of CONTEX T, a document production system, based on $\text{T}_\text{E}_\text{X}$.

CONTEX T offers the user a flexible and high quality typesetting environment. No in-depth knowledge of $\text{T}_\text{E}_\text{X}$ is needed. The parameter driven character of CONTEX T enables users to define their own layout rather easily.

CONTEX T is developed and tested in a production environment and is used for typesetting simple books as well as complex documents, paper and/or screen based. This introduction manual describes the functionality needed for everyday publications, like manuals and educational materials.

This manual is also available as an interactive document, be it in a bit different layout. The macro package CONTEX T, some more advanced examples and additional information can be found at www.pragma-ade.com.

PRAGMA ADE
Ridderstraat 27
8061 GH Hasselt NL
www.pragma-ade.com

\backslash about \backslash adaptlayout \backslash appendix \backslash at \backslash atpage \backslash background \backslash blackrule \backslash blackrules \backslash blank \backslash bookmark \backslash but \backslash button \backslash bypassblocks \backslash CAP
 \backslash Cap \backslash Caps \backslash Character \backslash Characters \backslash cap \backslash chapter \backslash character \backslash characters \backslash chem \backslash clip \backslash clonefield \backslash color \backslash column \backslash comment
 \backslash comparecolorgroup
 \backslash couplepage \backslash co
 \backslash definebodyfont
 \backslash definedfont \backslash de
 \backslash definefontssynonym
 \backslash definelist \backslash def
 \backslash defineprogram
 \backslash definestartstop
 \backslash definetyping
 \backslash fieldstack \backslash fi
 \backslash footnote \backslash footn
 \backslash headnumber \backslash hea
 \backslash installlanguage
 \backslash load* \backslash logfield
 \backslash midaligned \backslash mir
 \backslash nop \backslash nospace
 \backslash part \backslash periods
 \backslash placelogos \backslash pla
 \backslash processblocks
 \backslash reset \backslash resetma
 \backslash selectversion
 \backslash setupbodyfont
 \backslash setupclipping
 \backslash setupexternalfigu
 \backslash setupfontssynonym
 \backslash setuphead \backslash set
 \backslash setupinmargin
 \backslash setupitemgroup
 \backslash setuplistalternat
 \backslash setupoutput
 \backslash setupparagraphs
 \backslash setupregister
 \backslash setupsymbolset
 \backslash setupextrules
 \backslash setupversions
 \backslash showgrid \backslash show
 \backslash startalignment
 \backslash startenvironment
 \backslash startlines \backslash st
 \backslash startopposite
 \backslash startquotation
 \backslash stretched \backslash sub
 \backslash tab \backslash tex \backslash t
 \backslash underbars \backslash use
 \backslash useexternaldocument \backslash useexternalfigure \backslash useexternalfile \backslash useexternalfiles
 \backslash useexternalsoundtrack \backslash usemodule \backslash useresources \backslash usespecials \backslash usesymbols \backslash usetypescript \backslash usetypescriptfile \backslash useURL \backslash version \backslash v1 \backslash WEEKDAY
 \backslash WORD \backslash WORDS \backslash Word \backslash Words \backslash weekday \backslash whitespace \backslash wordright \backslash writebetweenlist \backslash writetolist \backslash writetoreferencelist \backslash writetoregister \backslash *

convertnumber \backslash copyfield \backslash correctwhitespace \backslash coupledocument \backslash coupled* \backslash couplemarking
e \backslash currentheadnumber \backslash date \backslash decouplemarking \backslash define \backslash defineblank \backslash defineblock
incolor \backslash definecolorgroup \backslash definecombinedlist \backslash defineconversion \backslash definedescription
 \backslash definefiguresymbol \backslash definefloat \backslash definefont \backslash definefontfeature \backslash definefonthandling
l \backslash defineindentedtext \backslash defineinteractionmenu:1 \backslash defineinteractionmenu:2 \backslash definelabel
utput \backslash defineoverlay \backslash definepalet \backslash definepapersize \backslash defineparagraphs \backslash defineprofile
nereferencelist \backslash defineregister \backslash definesection \backslash definesectionblock \backslash definesorting
ns \backslash definetabletemplate \backslash definetabulate \backslash definetext \backslash definetype \backslash definetypeface
elistscharacteristics \backslash disableinteractionmenu \backslash emptylines \backslash externalfigure \backslash field
:ext \backslash fitfield \backslash fixedspaces \backslash followprofile \backslash followprofileversion \backslash followversion
 \backslash getbuffer \backslash getmarking \backslash godown \backslash goto \backslash gotobox \backslash graycolor \backslash grid \backslash hairline \backslash head
 \backslash inframed \backslash ininner \backslash inleft \backslash inline \backslash inmargin \backslash inothmargin \backslash inouter \backslash inright
items \backslash its \backslash keepblocks \backslash labels \backslash labeltext \backslash language \backslash leftaligned \backslash listsymbol
:size \backslash mar \backslash marginrule \backslash marginrule \backslash marking \backslash markversion \backslash mathematics \backslash mediaeval
 \backslash noheaderandfooterlines \backslash noindenting \backslash noindex \backslash nomarking \backslash nomoreblocks \backslash nomorefiles
nbers \backslash overbar \backslash overbars \backslash overstrike \backslash overstrikes \backslash packed \backslash page \backslash pagereference
otnotes \backslash placeformula \backslash placelegend \backslash placelist \backslash placelistof* \backslash placelocalfootnotes
list \backslash placeregister:1 \backslash placesidebyside \backslash placesubformula \backslash place* \backslash place:2* \backslash position
 \backslash quote \backslash Romannumerals \backslash ran \backslash ref \backslash reference \backslash referral \backslash referraldate \backslash reserve*
nnumerals \backslash rotate \backslash scale \backslash screen \backslash section \backslash see* \backslash selectblocks \backslash selectpaper
 \backslash setupbackground \backslash setupbackgrounds \backslash setupblackrules \backslash setupblank \backslash setupblock
ottomtexts \backslash setupbuffer \backslash setupbuttons \backslash setupcapitals \backslash setupcaption \backslash setupcaptions
combinations \backslash setupcombinedlist \backslash setupcomment \backslash setupdescriptions \backslash setupenumerations
s \backslash setupfillinrules \backslash setupfloat \backslash setupfloats \backslash setupfloatsplitting \backslash setupfonthandling
definition \backslash setupfootnotes \backslash setupforms \backslash setupformulas \backslash setupframed \backslash setupframedtexts
 \backslash setupheads \backslash setupheadtext \backslash setuphyphenmark \backslash setupindentedtext \backslash setupindenting
interactionmenu \backslash setupinteractionscreen \backslash setupinterlinespace:1 \backslash setupinterlinespace:2
setuplayout \backslash setuplegend \backslash setuplinenumbering \backslash setuplines \backslash setuplinewidth \backslash setuplist
pmarginrules \backslash setupmarking \backslash setupnarrower \backslash setupnumbering \backslash setupoppositeplacing
getransitions \backslash setappalet \backslash setuppaper \backslash setuppapersize \backslash setupparagraphnumbering
programs \backslash setuppublications \backslash setupquote \backslash setupreferencelist \backslash setupreferencing
 \backslash setupsectionblock \backslash setupsorting \backslash setupspacing \backslash setupstrut \backslash setupsubpagenumber
 \backslash setupsynonyms \backslash setupsystem \backslash setuptab \backslash setuptables \backslash setuptabulate \backslash setuptext
ice \backslash setuptop \backslash setuptoptexts \backslash setuptype \backslash setuptyping \backslash setupunderbar \backslash setupurl
vironment \backslash showcolor \backslash showcolorgroup \backslash showexternalfigures \backslash showfields \backslash showframe
 \backslash showsetups \backslash showstruts \backslash showsymbolset \backslash someline \backslash somewhere \backslash space \backslash splitfloat
 \backslash startcolumns \backslash startcombination \backslash startcomment \backslash startcomponent \backslash startdocument
rtinteractionmenu \backslash startlegend \backslash startline \backslash startlinecorrection \backslash startlinenumbering
notes \backslash startmakeup \backslash startmakeup* \backslash startmarginblock \backslash startmarginrule \backslash startnarrower
 \backslash startpositioning \backslash startpostponing \backslash startproduct \backslash startprofile \backslash startproject
ole \backslash starttables \backslash startextrule \backslash starttext* \backslash startunpacked \backslash startversion \backslash start*
ion \backslash subsubsubject \backslash switchtobodyfont \backslash sym \backslash symbol \backslash synchronizationbar \backslash synchronize
 \backslash title \backslash tooltip \backslash translate \backslash typ \backslash type \backslash typebuffer \backslash typefile \backslash underbar