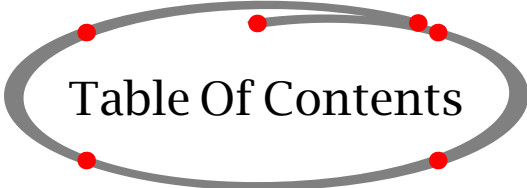


# Getting Started with CONTEXT

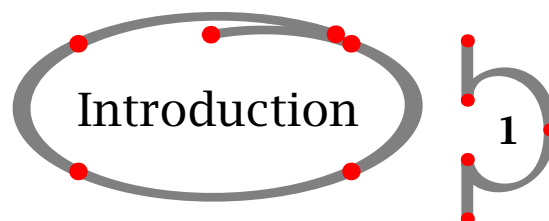
English  
Version

Ton Otten & Hans Hagen  
PRAGMA ADE  
maandjaar



# Table Of Contents

Introduction	2	23 Registers	55
1 How to create a document	3	24 Synonyms	56
2 How to proces a file / Run CONTEX	5	25 Referring to text elements	57
3 Special characters	6	26 Color	60
4 Defining a document	7	27 Background of text	61
5 Headers	8	28 Background of page areas	61
6 Itemize	10	29 Alignment	62
7 Formulas	15	30 Interactive mode in electronic documents	63
8 Legends	17	31 Fonts and fontswitches	67
9 Units	18	32 Composed characters	72
10 Figures	20	33 Page layout	73
11 Tables	25	34 Paragraph spacing	77
12 Paragraph formatting	36	35 Setup commands	81
13 Columns	39	36 Defining commands / macros	82
14 Footnotes	41	37 Miscellaneous	84
15 Definitions	42	38 Using modules	94
16 Numbered definitions	43	A User specifications	96
17 Outlined text	45	B Processing steps	96
18 Outlined paragraphs	46	C Auxilliary files	96
19 Margin texts	47	D Command definitions	98
20 Page breaking and page numbering	48	E Command Index	123
21 Page headers and footers	50	F Subject Index	126
22 Table of contents (lists)	52		



CONTEX<sub>T</sub> is a document engineering system and based on T<sub>E</sub>X. T<sub>E</sub>X is a typesetting system and a program to typeset and produce documents. CONTEX<sub>T</sub> is easy to use and enables you to make complex paper and electronic documents.

This manual describes the capabilities of CONTEX<sub>T</sub> and the available commands and their functionality.<sup>1</sup>

CONTEX<sub>T</sub> is developed for practical applications: the typesetting and production of documents ranging from simple straight forward books upto very complex and advanced technical manuals and studybooks in a paper or electronic version. This introductory manual describes CONTEX<sub>T</sub> functionality necessary to apply standard text elements in a manual or studybook. CONTEX<sub>T</sub> however is capable of much more and for users who want more there are other CONTEX<sub>T</sub> manuals and sources available.

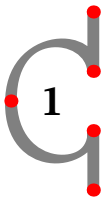
CONTEX<sub>T</sub> has a multilingual interface to enable users to work with CONTEX<sub>T</sub> in their own language. CONTEX<sub>T</sub> and this manual is available in dutch, german and english.

CONTEX<sub>T</sub> is published in the public domain with the help of the Nederlandstalige T<sub>E</sub>X Gebruikersgroep (NTG). All CONTEX<sub>T</sub> products and information can be obtained form the NTG-server ([WWW.NTG.NL](http://WWW.NTG.NL)).

---

<sup>1</sup> All paper and electronic products around CONTEX<sub>T</sub> are produced with CONTEX<sub>T</sub>. All sources of these products are or will be made available electronically to give you insight in the way these products are made up.





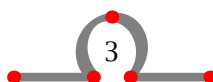
# 1 | How to create a document

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

CONTEXT 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 `myfile.tex` you will find no difficulties in running CONTEXT.

An input file could look like this:

```
\starttext
\startstandardmakeup
  \middlelined{How to make a document.}
  \middlelined{by}
  \middlelined{The Author}
\stopstandardmakeup
\completecontent
\chapter{Introduction}
... your text\index{indexentry} ...
\chapter{One Chapter}
\section[firstparagraph]{The first paragraph}
... your text ...
\section{The second paragraph}
\subsection{the first sub paragraph}
... your text\index{another indexentry} ...
\subsection{the second sub paragraph}
... your text ...
\section{The third paragraph}
... your text ...
\chapter{Another Chapter}
... your text ...
\chapter[Lastchapter]{The Last Chapter}
```



```
... your text ...  
\completeregister  
\stoptext
```

CONTEX<sub>T</sub> expects a plain ASCII input file. Off course you can use any texteditor or wordprocessor you want, but you should not forget that CONTEX<sub>T</sub> can only read ASCII input. Most texteditors or wordprocessors can export your file as plain ASCII.

The input file should contain the text you want to be processed by CONTEX<sub>T</sub> and the CONTEX<sub>T</sub> commands. A CONTEX<sub>T</sub> command begins with a backslash \. With the command \starttext you indicate the beginning of your text. The area before \starttext is called the set up area and is used for defining new commands and set up the layout of your document.

A command is mostly followed by a left and right bracket pair [] and/or a left and right brace {}. In \chapter[*lastchapter*]{The Last Chapter} the command \chapter for example tells CONTEX<sub>T</sub> to perform a few actions concerning design, typography and structure. These actions might be:

1. start with 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 table of contents

These actions will be performed on the argument that is given between the left and right braces: *The Last Chapter*.

Thusfar the [*lastchapter*] between brackets was not mentioned. This is a label with a logical name that can be used for refering to that specific chapter. This can be done with yet another CONTEX<sub>T</sub> command: \in{chapter}[*lastchapter*].

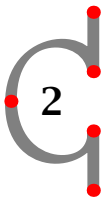
So now the list of actions can be extended with:

7. let label *lastchapter* be chapter number (and store this for later use)

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

If you have CONTEX<sub>T</sub> process this file, you would obtain a very simple document with only a few numbered chapter headers and paragraph headers.

While processing this file CONTEX<sub>T</sub> takes care of many things. One of these things is for example page numbering. But in order to make a table of contents CONTEX<sub>T</sub> needs page numbers that are not known to CONTEX<sub>T</sub> yet at the first run. So you have to proces this file twice (a two pass job). CONTEX<sub>T</sub> will produce a few auxilliary files to store this kind of information. In some instances you have to proces an input file thrice (a three pass job).



## 2 | How to proces a file / Run `CONTEXT`

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

```
ctex filename
```

The availability of the batch command `ctex` depends on the system you're using. Ask your system administrator what with what command you can start `CONTEXT`. If your filename is `myfile.tex` this can be:

```
ctex myfile
```

the extension `.tex` is not needed.

After typing `Enter` processing will be started. `CONTEXT` will show processing information on your screen. If processing is succesful the command line prompt will return and `CONTEXT` will produce a `dvi` file.

If processing is not succesful —for example because you typed `stptext` instead of `\stoptext`— `CONTEXT` produces a `?` on you terminal 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 <code>CONTEXT</code> 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.

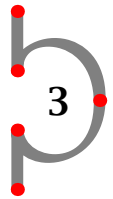
Some errors will produce a `*` on your screen and processing will stop. This error is due to a fatal error in your input file. You can't ignore this error and the only option you have is to type `\stop` or `Ctrl Z`. The program will be halted and you can fix the error.

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 proces steps. Further more it gives of 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.



## Special characters

When processing is successful `CONTEXT` has produced a new file, with the extension `.dvi`. In case of running `myfile.tex` the file `myfile.dvi` is produced. The abbreviation `dvi` stands for Device Independent. In this case this means that the file can be processed by a printerdriver to make the file suitable for printing. The `dvi` file can be transformed to a `ps` file, `HP` file or whatever file you want.



You have seen that `CONTEXT` commands are preceded by a `\` (backslash). This means that `\` has a special meaning to `CONTEXT`. Next to `\` 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.

Special character	Verbatim		Text	
	Type	To produce	Type	To produce
#	<code>\type{#}</code>	#	<code>\#</code>	#
\$	<code>\type{\$}</code>	\$	<code>\\$</code>	\$
&	<code>\type{&amp;}</code>	&	<code>\&amp;</code>	&
%	<code>\type{%}</code>	%	<code>\%</code>	%

Table 3.1 Special characters (1).

Other special characters have a meaning in typesetting mathematical expressions and some can be used in math mode only (see chapter 7).

Special character	Verbatim		Text	
	Type	To produce	Type	To produce
+	<code>\type{+}</code>	+	<code>+\$</code>	+
-	<code>\type{-}</code>	-	<code>-\$</code>	-
=	<code>\type{=}</code>	=	<code>=\$</code>	=
<	<code>\type{&lt;}</code>	<	<code>&lt;\$</code>	<
>	<code>\type{&gt;}</code>	>	<code>&gt;\$</code>	>

Table 3.2 Special characters (2).



## 4 | Defining a document

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.

Set ups are typed in the set up area just before `\starttext`.

```
\setupcorps[12pt]
\starttext
This is a one line document.
\stoptext
```

Within the `\starttext ... \stoptext` a document can be divided in three sections:

1. introductions
2. main texts
3. extroductions
4. appendices

The sections are defined with:

```
\startintroductions ... \stopintroductions
\startmaintexts      ... \stopmaintexts
\startextroductions ... \stopextroductions
\startappendices    ... \stopappendices
```

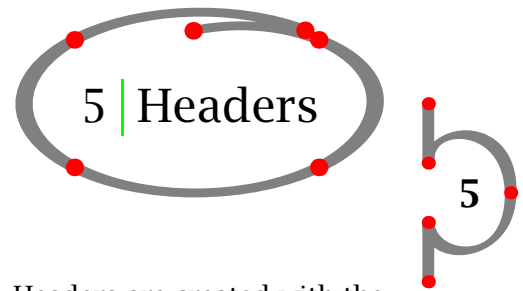
In the introduction as well as extroduction section the command `\chapter` produces a none numbered header in the table of contents. This section is mostly used for the table of contents, the list of figures and tables, the preface, the acknowledgements etcetera. This section often has a roman page numbering.

The appendices section is used for (indeed) appendices. Headers can be typeset in a different way. For example `\chapter` can be numbered alphabetically.

The sections can be set up with:

```
\setupsectionblock[...][...,...=...,...]
```





The structure of a document is determined by its headers. Headers are created with the commands shown in table 5.1:

Numbered header	Numberless 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 5.1 Headers.

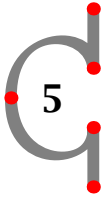
```
\chapter[ref,...]{...}
```

```
\section[ref,...]{...}
```

```
\subsection[ref,...]{...}
```

```
\title[ref,...]{...}
```

```
\subject[ref,...]{...}
```



```
\subsubject[ref,...]{...}
```

These commands will produce a number and a header in a predefined fontsize and font-type with an amount of vertical spacing before and after the header.

The heading commands have two appearances. For example:

```
\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 header you type for example `\on{page}[hasselt-by-night]`.

Of course these headers can be set at your own preferences and you can even define your own headers. This is done by the command `\setuphead` and `\definehead`.

```
\setuphead[...] [..., ...=..., ...]
```

```
\definehead[.1.][.2.]
```

```
\definehead
```

```
  [myheader]
```

```
  [section]
```

```
\setuphead
```

```
  [myheader]
```

```
  [numberstyle=bold,
```

```
  textstyle=kap,
```

```
  before=\hairline\blank,
```

```
  after=\nowhitespace\hairline]
```

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

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

---

## 5.1 HASSELT MAKES HEADLINES

---

## Itemize

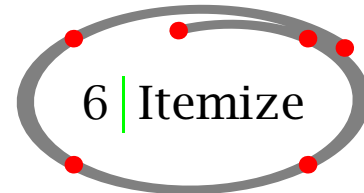
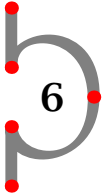
There is one other command you should know and that is: `\setupheads`. You can use this command to set up the numbering of the numbered headers. If you type:

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

all numbers will appear in the margin. Paragraph 1.1 would look as 1-1.

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

```
\setupheads[...,...=...,...]
```



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

```
\startitemize[...,...][...,...=...,...] ... \stopitemize
```

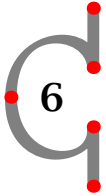
For example:

```
\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 `\startitemize ... \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 vertical spacing after the item separator. 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.

Itemization is a two pass job. This means that you have to run your file twice for an optimal layout. The bracket pair contains information on item separators and local set up variables.



Argument	Item separator
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 6.1 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.

Sometimes you would like to have headers within an enumeration. In that case you should use `\head` instead of `\item`.

```
Hasselt lies in the province of Overijssel and there are a number of
customs that are typical for 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) if he will become a strong man.

## Itemize

The neighbours will bring a *krentenwegge* along. A *krentenwegge* is a loaf of currant bread of about 1 \Meter\ long. Off course the birth is celebrated with *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 *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 *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 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 for 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) if he will become a strong man. The neighbours will bring a *krentenwegge* along. A *krentenwegge* is a loaf of currant bread of about 1 m long. Off course the birth is celebrated with *jenever*.

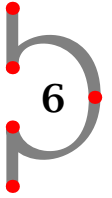
- nabuurschap (naberschap) (neighbourship)

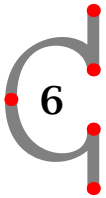
Smaller communities used to be very dependent on the cooperation among the members for their well being. Members of the *nabuurschap* helped each other in difficult times during harvest times, funerals or any hardship that fell upon the community.

- 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 are described in table 6.2.





Set up	Meaning
standard	standard set up (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 6.2 Set up parameters in 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 | 4. Eikenlaan      | 7. Heerengracht | 10. Julianakade     |
| 2. Baangracht     | 5. Eiland         | 8. Hofstraat    | 11. Justitiebastion |
| 3. Brouwersgracht | 6. Gasthuisstraat | 9. Hoogstraat   |                     |

Sometimes you want to continue the enumeration after a short intermezzo. Then you type for example `\startitemize[continue,columns,three,broad]` and numbering will continue in a three column format.

- |                  |                    |                    |
|------------------|--------------------|--------------------|
| 12. Kaai         | 17. Meestersteeg   | 22. Royenplein     |
| 13. Kalverstraat | 18. Prinsengracht  | 23. Van Nahuijsweg |
| 14. Kastanjelaan | 19. Raamstraat     | 24. Vicariehof     |
| 15. Keppelstraat | 20. Ridderstraat   | 25. Vissteeg       |
| 16. Markt        | 21. Rosmolenstraat | 26. Watersteeg     |

## Itemize

27. Wilhelminalaan

28. Ziekenhuisstraat

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

```
\setupitemize[.1.][...2...][...,...=...]
```

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

```
In The Netherlands the cities can determine the height of a number of taxes. So the costs of living can differ from town to town. There are differences of upto 50\% in taxes like:
```

```
\setupitemize[2][width=5em]
```

```
\startitemize[n]
```

```
\item real estate tax
```

```
    The real estate tax is divided in 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 cityhall.
```

```
\stopitemize
```

then the horizontal space between item separator and text at the second level 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 costs of living can differ from town to town. There are differences of upto 50% in taxes like:
```

```
1. real estate tax
```

```
    The real estate tax is divided in two components:
```

```
    a.      the ownership tax
```

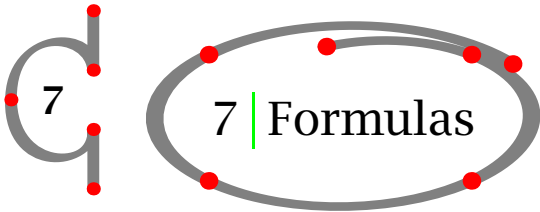
```
    b.      the tenant tax
```

```
    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 cityhall.



You can typeset numbered formulas with:

```
\placeformula[ref,...].1.$$2.$$
```

```
\startformula ... \stopformula
```

Two examples:

```
\placeformula[formula:aformula]
\startformula
  y=x^2
\stopformula

\placeformula
\startformula
  \int_0^1 x^2 dx
\stopformula
```

$$y = x^2 \tag{7.1}$$

$$\int_0^1 x^2 dx \tag{7.2}$$

Between `\startformula` and `\stopformula` you are in math mode so you can define any formula you want by using  $\TeX$  commands. We advise you to have some further reading on typesetting formulas in  $\TeX$ . See for example:

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



## Formulas

Math mode can occur in two modes: in text mode and display mode. Mathematic expressions in text mode are placed between `$` and `$`.

The Hasselt community covers an area of 42,05 \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 `$$$ \frac{1}{4} \pi r^2$`.

This will become:

The Hasselt community covers an area of 42,05 km<sup>2</sup>. 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 `{}` in  $\frac{1}{4}\pi r^2$  are essential for separating operations in the expression. If you omit the outer curly braces like this: `$$$ \frac{1}{4} \pi r^2$`, you would get a non desired result:  $\frac{1}{4\pi r^2}$ .

Display mode however begins with `$$$` and ends with `$$$`.

If you type:

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

you will get an expression that is *displayed* in the middle of a page.

$$\int_0^1 x^2 dx$$

If you compare `$$$` with the first examples you'll notice that `\startformula` and `\stopformula` is equivalent to the `$$$`.

The command `\placeformula` handles spacing around the formulas and the numbering of the formula. The bracket pair is optional and is used for referring and switching numbering off. The curly braces are also optional and are used for text.

$$y = x^2 \tag{7.3}$$

$$y = x^3 \tag{7.4}$$

$$y = x^4 \tag{7.5}$$

Formula 7.4 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]`.

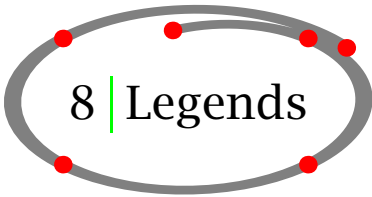
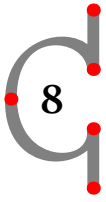
If no numbering is required you type:

`\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, intermezzos etcetera 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:

```
\setupformulas[...]=...]
```



The commands `\startlegend ... \stoplegend` are used to explain the meaning of symbols in formulas. For example, if you want to display a formula and its legend:

$$s = vt + \frac{1}{2}at^2 \tag{8.1}$$

- $s$  = displacement m
- $v$  = velocity m/s
- $t$  = time s
- $a$  = acceleration m/s<sup>2</sup>

You will have to type something like this:

```
\placeformula$$ s = v t + \frac{1}{2} a t^2 $$
\startlegend
\leg s \ displacement \ m \
\leg v \ velocity \ m/s \
\leg t \ time \ s \
```

## Units

```
\leg a \\ acceleration \\ m/s^2 \\  
\stoplegend
```

The command `\leg` starts a new row and `\\` is a column separator. The last `\\` in each row is essential. The spaces are optional and have no effect on the outcome. However, these commands are easily mistyped and misread while checking. So take some time to make a readable input file.

```
\startlegend[.1.]\leg.2.\\.3.\\.4.\\\leg ... \stoplegend
```

The first column is used for symbols and is typeset in mathematical mode. The second column is used for explanations of the symbols and the third one for units and dimensions. Because of the necessity of sub- and superscript the third column is in mathematical mode, but the text is displayed in the actual font.

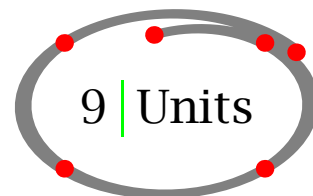
Complementary to legends is a command that displays facts in a consistent way when formulas are used for calculations.

Such a command looks like this:

```
\startfact  
\fact velocity    \\ v \\ 10~m/s  \\  
\fact acceleration \\ a \\ - 3~m/s^2 \\  
\fact time        \\ t \\ 3~s     \\  
\stopfact
```

The two last columns are in mathematical mode, the last column displays text in the actual font. The `~` is obligatory, because spaces are ignored in mathematical mode.

```
velocity    v = 10 m/s  
acceleration a = -3 m/s2  
time        t = 3 s
```



To force yourself to use dimensions and units consistently throughout the document you can make your own list with units. These are specified in the set up area of your input file.

In `CONTEXT` there is an external module available that contains almost all SI-units. When this module is loaded with `\usemodule[unit]` you can call units with:

```
\startlines
\Meter \Per \Square \Meter
\Kubic \Meter \Per \Sec
\Square \Milli \Meter \Per \Inch
\Centi \Liter \Per \Sec
\Meter \Inverse \Sec
\Newton \Per \Square \Inch
\Newton \Times \Meter \Per \Square \Sec
\stoplines
```

It looks like a lot of typing but it does guarantee a consistent use of units. The command `\unit` also prevent the separation of value and unit at line breaking. The unit would come out as:

```
m/m2
m3/s
mm2/inch
cl/s
m s-1
N/inch2
N · m/s2
```

You can define your own units with:

```
\unit[Ounce]{oz}{}
```

Later on in the document you can type `15.6 \Ounce` that will be displayed as 15.6 oz.

The legend on page 17 would be more consistent when you type it like this:

```
\startlegenda
\leg s \ place           \ \Meter                \
\leg v \ velocity       \ \Meter \Per \Sec        \
\leg t \ time           \ \Sec                    \
\leg a \ acceleration   \ \Meter \Per \Square \Sec \
\stoplegenda
```

The command `\unit` is an application of synonyms. See chapter 24 for more information on synonyms.

In order to write  $\%$  and  $\text{‰}$  in a consistent way there are two specific commands:

```
\procent
\promille
```

# 10 | Figures

Photographs and pictures can be inserted in your document with the following command:

```
\placefigure  
  [] [fig:church]  
  {Stephanus Church.}  
  {\name{hass24g}}
```

After processing this would be:

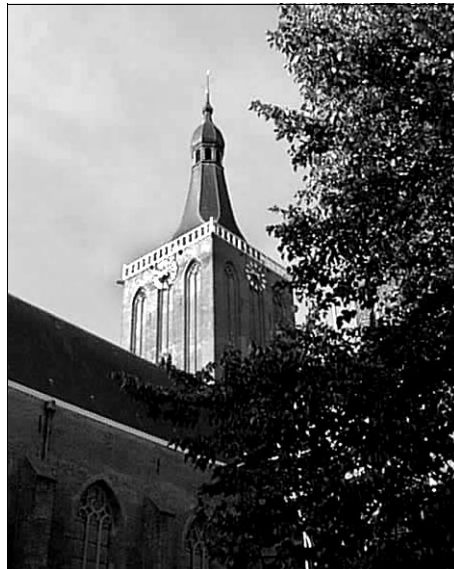


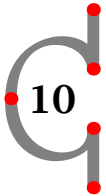
Figure 10.1 Stephanus Church.

The command `\placefigure` handles numbering and vertical spacing before and after your figure. Furthermore this command initializes a floatmechanism. This means that `CONTEXT` looks whether there is enough space for your figure. If not the figure will be placed at another location while open space is filled out with text: the figure starts floating in your document until the most optimal location is found. You can influence this mechanism within the first bracket pair.

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

```
\placeblock[.1.][ref,...]{.2.}{.3.}
```

The options are described in table 10.1.



Option	Meaning
here	put figure at this location if possible
force	ignore float mechanism and place figure
page	put figure at top of the next page
top	put the caption above figure
bottom	put caption under figure
left	place figure at the left margin
right	place figure at the right margin

Table 10.1 Options in `\placefigure`.

The second bracket pair is used for refering. 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. If you want no caption and no number, you can type `{none}`. The figure labels are set up with `\setupcaption` and the numbering is (re)set by `\setupnumbering` (see paragraph 37.1).

The second brace pair is used for defining the figure or adresssing 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
  {A framed Hasselt.}
  {\framed{\tfd Hasselt}}
```

This will produce:



Figure 10.2 A framed Hasselt.

Most of the time however your pictures are made up in a macropackages like Corel Draw or Illustrator and photos are — after scanning — improved in packages like PhotoShop.

## Figures

Next the figures are available as files. `CONTEXT` in conjunction with `TEXUTIL` supports default the `eps` file format.<sup>2</sup> Therefore you should convert your `tif` files to `eps`. Before processing your input file `CONTEXT` needs some information on the figures like file format and dimensions. This is done by `\useexternalfigure`.

In figure 25.1 you see a photo and a graphic combined into one figure.



a bitmap picture



a vector graphic

Figure 10.3 The Hasselt Canals.

You can produce this figure by typing something like:

```
\useexternalfigure
  [Photo][hass03g]
  [type=tif,width=.4\textwidth]
\useexternalfigure
  [Graphic][gracht]
  [type=eps,width=.4\textwidth]

\placefigure
  [here,force]
  [fig:canal]
  {The Hasselt Canals.}
  \startcombinatie[2*1]
    {\Photo}{a bitmap picture}
    {\Graphic}{a vector graphic}
  \stopcombinatie
```

```
\useexternalfigure[.1.][.2.][...,...=,...]
```

<sup>2</sup> Some `dvi` to `ps` drivers support the `tif` format. This manual for example is produced with `DVIPSONE`.

## Figures

The command `\useexternalfigure` has three bracket pairs. The first pair contains a logical name of the figure. This is optional and used when one figure appears more than once in your document.

The second pair contains the file name (without extension). The third pair leaves room for filetype definition and some set up parameters. In well structured documents you would type `\useexternalfigure` in the set up area of your input file.

10

```
\startcombination[...] ... \stopcombination
```

The `\startcombination ... \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]` (`[h*v]`).

The examples shown above are enough for creating illustrated documents. Sometimes however you want a more integrated layout or the picture and the text. For that purpose you can use:

```
\startblocktext[.1.][ref]{.2.}{.3.} ... \stopblock
```

The effect of:

```
\startfiguretext
[left]
[fig:citizens]
{none}
{\externalfigure[hass07g]
[type=tif,width=.5\makeupwidth,frame=on]}
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 touristic attraction.
\stopfiguretext
```

is shown in the figure below.



## Figures



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 touristic attraction.

10

```
\externalfigure[...][...,...=...,...]
```

The last curly brace pair enclose the command `\externalfigure`. This command gives you the freedom to do anything you want with a figure. `\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:<sup>3</sup>

```
\inmarge
  {\externalfigure
   [hass23g]
   [type=tif,width=\marginwidth,frame=on]}
```

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.

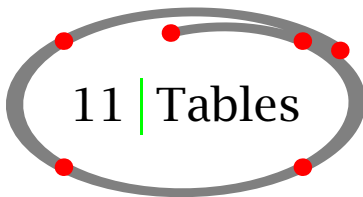
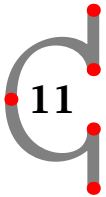
```
\setupfloats
  [location=right]
\setupcaptions
```

<sup>3</sup> See page 48

## Tables

```
[location=top,  
height=.4\makeupheight,  
character=boldslanted]  
  
\placefigure  
{Just a picture.}  
{\externalfigure[hass18g][frame=on]}
```

Figure 10.4 Just a picture.



*In general, a table consists of columns which may be independently left adjusted, centered, right adjusted, or aligned on decimal points. Headings may be placed over single columns or groups of columns. Table entries may contain equations or several rows of text. Horizontal and vertical lines may be drawn wholly or partially across the table.*

This is what Michael J. Wichura wrote in the preface of the manual of `TABLE` (`TABLE` manual, 1988.). Michael Wichura is also the author of the `TABLE` macros `CONTEXT` is relying on when processing tables. A few `CONTEXT` macros were added to take care of consistent line spacing and to make the interface a little less cryptic.<sup>4</sup>

## Tables

For placing a table the command `\placetable` is used which is a predefined version of:

```
\placeblock[.1.][ref,...]{.2.}{.3.}
```

For defining the table you use:

```
\starttable[...] ... \stoptable
```

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
```

And after processing the table would be:

<b>Year</b>	<b>Number of ships</b>
1645	450
1671	480
1676	500
1695	930

Table 11.1 Ships that moored at Hasselt.

The first command `\placetable` has the same function as `\placefigure`. It takes care of spacing before and after the table and numbering. Furthermore the floating mechanism is initialized so the table will be placed at the most optimal location of the page.

<sup>4</sup> `CONTEXT` was developed for non-technical users in the WYSIWYG era. Therefore a user friendly interface and easy file and command handling were evident and cryptic commands, programming and logical reasoning had to be avoided.

The table entries are placed between the `\starttable ... \stoptable` pair. Between the bracket pair you can specify the table format with the column separators `|` and the format keys (see table 11.2).

Key	Meaning
	column separator
c	center
l	flush left
r	flush right
s<n>	set intercolumn space at value $n = 0, 1, 2$
w<n>	set minimum column width at specified value

Table 11.2 Table format keys.

In addition to the format keys there are format commands. Table 11.3 shows a few of the essential commands.

Command	Meaning
<code>\JustLeft</code>	flush left and omit column format
<code>\JustRight</code>	flush right and omit column format
<code>\JustCenter</code>	center and omit column format
<code>\SetTableToWidth{}</code>	specify exact table width
<code>\use{n}</code>	use the space of the next $n$ columns

Table 11.3 Table format commands.

In the examples you have seen so far a number of `CONTEXT` formatting commands were used. These commands are somewhat longer than the original and less cryptic but they also handle a lot of table typography. In table 11.4 an overview of these commands is given.

The tables below are shown with their sources. You can always read the `TABLE` manual by M.J. Wichura for more sophisticated examples.

```
\placetable
  [here,force]
  [tab:effects of commands]
  {Effect of formatting commands.}
\startcombination[2*1]
{\starttable[|c|c|]
\HL
```

## Tables

Command		Meaning
<code>\NR</code>	next row	make row with no vertical space adjustment
<code>\FR</code>	first row	make row, adjust upper spacing
<code>\LR</code>	last row	make row, adjust lower spacing
<code>\MR</code>	mid row	make row, adjust upper and lower spacing
<code>\SR</code>	separate row	make row, adjust upper and lower spacing
<code>\VL</code>	vertical line	draw a vertical line, go to next column
<code>\NC</code>	next column	go to next column
<code>\HL</code>	horizontal line	draw a horizontal
<code>\DL</code>	division line*	draw a division line over the next column
<code>\DL[n]</code>	division line*	draw a division line over $n$ columns
<code>\DC</code>	division column*	draw a space over the next column
<code>\DR</code>	division row*	make row, adjust upper and lower spacing
<code>\LOW{text}</code>	—	lower <i>text</i>
<code>\TWO, \THREE</code> etc.	—	use the space of the next <i>two, three</i> columns

\* `\DL`, `\DC` and `\DR` are used in combination.

Table 11.4 `CONTEXT` table format commands.

```

\VL \bf Year \VL \bf Citizens \VL\SR
\HL
\VL 1675 \VL ~428 \VL\FR
\VL 1795 \VL 1124 \VL\MR
\VL 1880 \VL 2405 \VL\MR
\VL 1995 \VL 7408 \VL\LR
\HL
\stoptable}{standard}
{\starttable[|c|c|]
\HL
\VL \bf Year \VL \bf Citizens \VL\NR
\HL
\VL 1675 \VL ~428 \VL\NR
\VL 1795 \VL 1124 \VL\NR
\VL 1880 \VL 2405 \VL\NR
\VL 1995 \VL 7408 \VL\NR
\HL
\stoptable}{only \type{\NR}}
\stopcombination

```

In the example above the first table `\SR`, `\FR`, `\MR` and `\LR` are used. These commands take care of line spacing within a table. As you can see below the command `\NR` only starts a new row.

## Tables

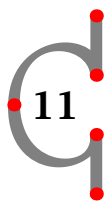
Year	Citizens
1675	428
1795	1124
1880	2405
1995	7408

standard

Year	Citizens
1675	428
1795	1124
1880	2405
1995	7408

only \NR

Table 11.5 Effect of formatting commands.



In the example below column interspacing with the s0 and s1 keys is shown.

```

\p[acetable
  [here, force]
  [tab:formatcommands]
  {Effect of formatting commands.}
\startcombination[3*2]
{\starttable[|c|c|]
\HL
\VL \bf Year \VL \bf Citizens \VL\SR
\HL
\VL 1675 \VL ~428 \VL\FR
\VL 1795 \VL 1124 \VL\MR
\VL 1880 \VL 2405 \VL\MR
\VL 1995 \VL 7408 \VL\LR
\HL
\stoptable}{standard}
{\starttable[s0 | c | c |]
\HL
\VL \bf Year \VL \bf Citizens \VL\SR
\HL
\VL 1675 \VL ~428 \VL\FR
\VL 1795 \VL 1124 \VL\MR
\VL 1880 \VL 2405 \VL\MR
\VL 1995 \VL 7408 \VL\LR
\HL
\stoptable}{\type{s0}}
{\starttable[| s0 c | c |]
\HL
\VL \bf Year \VL \bf Citizens \VL\SR
\HL
\VL 1675 \VL ~428 \VL\FR

```

## Tables

```

\VL 1795 \VL 1124 \VL\MR
\VL 1880 \VL 2405 \VL\MR
\VL 1995 \VL 7408 \VL\LR
\HL
\stoptable}{\type{s0} in column~1}
{\starttable[| c | s0 c |]
\HL
\VL \bf Year \VL \bf Citizens \VL\SR
\HL
\VL 1675 \VL ~428 \VL\FR
\VL 1795 \VL 1124 \VL\MR
\VL 1880 \VL 2405 \VL\MR
\VL 1995 \VL 7408 \VL\LR
\HL
\stoptable}{\type{s0} in column~2}
{\starttable[s1 | c | c |]
\HL
\VL \bf Year \VL \bf Citizens \VL\SR
\HL
\VL 1675 \VL ~428 \VL\FR
\VL 1795 \VL 1124 \VL\MR
\VL 1880 \VL 2405 \VL\MR
\VL 1995 \VL 7408 \VL\LR
\HL
\stoptable}{\type{s1}}
{}}
\stopcombination

```

After processing the tables would look something like this. The standard table means a column interspacing s2.

Columns are often separated with a vertical line | and rows by a horizontal line.

```

\placetable
[here,force]
[tab:divisions]
{Effect of options.}
\starttable[|c|c|c|]
\NC Steenwijk \NC Zwartsluis \NC Hasselt \NC\SR
\DC \DL \DC \DR
\NC Zwartsluis \VL Hasselt \VL Steenwijk \NC\SR
\DC \DL \DC \DR
\NC Hasselt \NC Steenwijk \NC Zwartsluis \NC\SR

```

Tables

Year	Citizens
1675	428
1795	1124
1880	2405
1995	7408

standard

Year	Citizens
1675	428
1795	1124
1880	2405
1995	7408

s0

Year	Citizens
1675	428
1795	1124
1880	2405
1995	7408

s0 in  
column 1

Year	Citizens
1675	428
1795	1124
1880	2405
1995	7408

s0 in column 2

Year	Citizens
1675	428
1795	1124
1880	2405
1995	7408

s1

Table 11.6 Effect of formatting commands.

Steenwijk	Zwartsluis	Hasselt
Zwartsluis	Hasselt	Steenwijk
Hasselt	Steenwijk	Zwartsluis

Table 11.7 Effect of options.

`\stoptable`

A more sensible example is given in the table below.

`\placetable`

`[here, force]`

`[tab:examplecontextcommands]`

`{Effect of \CONTEXT\ formatting commands.}`

`\starttable[|]|c|c|c|c|]`

`\HL`

`\VL \FIVE \JustCenter{Citycounsel elections in 1994} \VL\SR`

`\HL`

`\VL \LOW{Party} \VL \THREE{Districts} \VL \LOW{Total} \VL\SR`

`\DC \DL[3] \DC \DR`

`\VL \VL 1 \VL 2 \VL 3 \VL \VL\SR`

`\HL`



## Tables

```

\VL PvdA          \VL 351 \VL 433 \VL 459 \VL 1243      \VL\FR
\VL CDA           \VL 346 \VL 350 \VL 285 \VL ~981      \VL\MR
\VL VVD           \VL 140 \VL 113 \VL 132 \VL ~385      \VL\MR
\VL HKV/RPF/SGP  \VL 348 \VL 261 \VL 158 \VL ~767      \VL\MR
\VL GPV           \VL 117 \VL 192 \VL 291 \VL ~600      \VL\LR
\HL
\stoptable

```

In the last column a ~ is used to simulate a four digit number. The ~ has the width of a digit.

Citycounsel elections in 1994				
Party	Districts			Total
	1	2	3	
PvdA	351	433	459	1243
CDA	346	350	285	981
VVD	140	113	132	385
HKV/RPF/SGP	348	261	158	767
GPV	117	192	291	600

Table 11.8 Effect of CONTEXT formatting commands.

Sometimes your tables get too big and you want to adjust for example the korps or the vertical and/or horizontal spacing around vertical and horizontal lines. This is done by:

```
\setuptables[...]=...]
```

```

\placetable
  [here,force]
  [tab:setuptable]
  {Use of \type{\setuptables}.}
\startcombination[1*3]
{\setuptables[corps=10pt]
\starttable[|c|c|c|c|c|c|]
\HL
\VL \use6 \JustCenter{Decline of wealth in
                    Dutch florine (Dfl)} \VL\SR
\HL
\VL Year \VL 1.000--2.000

```

Tables

```

\VL 2.000--3.000
\VL 3.000--5.000
\VL 5.000--10.000
\VL over 10.000 \VL\SR

```

```

\HL
\VL 1675 \VL 22 \VL 7 \VL 5 \VL 4 \VL 5 \VL\FR
\VL 1724 \VL ~4 \VL 4 \VL -- \VL 4 \VL 3 \VL\MR
\VL 1750 \VL 12 \VL 3 \VL 2 \VL 2 \VL -- \VL\MR
\VL 1808 \VL ~9 \VL 2 \VL -- \VL -- \VL -- \VL\LR

```

```

\HL
\stoptable}{\tt corps=10pt}
{\setuptables[corps=8pt]
\starttable[|c|c|c|c|c|c|]
\HL
\VL \use6 \JustCenter{Decline of wealth in
Dutch florine (Df1)} \VL\SR

```

```

\HL
\VL Year \VL 1.000--2.000
\VL 2.000--3.000
\VL 3.000--5.000
\VL 5.000--10.000
\VL over 10.000 \VL\SR

```

```

\HL
\VL 1675 \VL 22 \VL 7 \VL 5 \VL 4 \VL 5 \VL\FR
\VL 1724 \VL ~4 \VL 4 \VL -- \VL 4 \VL 3 \VL\MR
\VL 1750 \VL 12 \VL 3 \VL 2 \VL 2 \VL -- \VL\MR
\VL 1808 \VL ~9 \VL 2 \VL -- \VL -- \VL -- \VL\LR

```

```

\HL
\stoptable}{\tt corps=8pt}
{\setuptables[corps=6pt,distance=small]
\starttable[|c|c|c|c|c|c|]
\HL
\VL \use6 \JustCenter{Decline of wealth in
Dutch florine (Df1)} \VL\SR

```

```

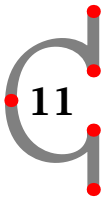
\HL
\VL Year \VL 1.000--2.000
\VL 2.000--3.000
\VL 3.000--5.000
\VL 5.000--10.000
\VL over 10.000 \VL\SR

```

```

\HL

```



## Tables

```

\VL 1675 \VL 22 \VL 7 \VL 5 \VL 4 \VL 5 \VL\FR
\VL 1724 \VL ~4 \VL 4 \VL -- \VL 4 \VL 3 \VL\MR
\VL 1750 \VL 12 \VL 3 \VL 2 \VL 2 \VL -- \VL\MR
\VL 1808 \VL ~9 \VL 2 \VL -- \VL -- \VL -- \VL\LR
\HL
\stoptable}{\tt corps=6pt,distance=small}
\stopcombination

```

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	-	-	-

**corps=10pt**

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	-	-	-

**corps=8pt**

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	-	-	-

**corps=6pt,distance=small**

Table 11.9 Use of `\setuptables`.

You can also set up the layout of tables with:

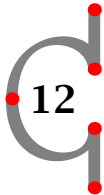
`\setupfloats[...]=...]`

You can set up the numbering and the labels with:

```
\setupcaption[...],[...,...=...,...]
```

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]
\setupcaption[character=boldslanted]
\placetable{Opening hours library.}
\starttable[|l|c|c|]
\HL
\VL \bf Day \VL \use2 \bf Opening hours \VL\SR
\HL
\VL Monday \VL 14.00 -- 17.30 \VL 18.30 -- 20.30 \VL\FR
\VL Tuesday \VL \VL \VL \VL\MR
\VL Wednesday \VL 10.00 -- 12.00 \VL 14.00 -- 17.30 \VL\MR
\VL Thursday \VL 14.00 -- 17.30 \VL 18.30 -- 20.30 \VL\MR
\VL Friday \VL 14.00 -- 17.30 \VL \VL\MR
\VL Saturday \VL 10.00 -- 12.30 \VL \VL\LR
\HL
\stoptable
```



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	

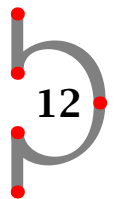
Table 11.10 Opening hours library.

## 12 | Paragraph formatting

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

```
\defineparagraphs[...][...,...=...,...]
```

```
\setupparagraphs[.1.][.2.][...,...=...,...]
```



In the example below is shown how to typeset complete paragraphs columnwise.

```
\defineparagraphs
  [mypar] [n=3, before={\blank}, after={\blank}]

\setupparagraphs
  [mypar] [1] [width=.1\textwidth, character=bold]
\setupparagraphs
  [mypar] [2] [width=.4\textwidth]

\startmypar
1252
\mypar
Hasselt obtains its city charter from bishop Hendrik van Vianden.
\mypar
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.
\stopmypar

\startmypar
1350
\mypar
Hasselt joins the Hanzepact to protect their international trade.
\mypar
The Hanzepact was of great importance for merchants in Hasselt. In
those days trading goods were taxed at every city, highway or
```

## Paragraph formatting

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.

`\stopmypar`

After defining a paragraph with `\defineparagraphs` you can format the paragraph with `\setupparagraphs`. Next you can start your paragraph with `\start...` and end it with `\stop...`. A new paragraph starts with the name of your paragraph, in this case `\mypar`.

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 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.

Another example of paragraph formatting could look like this.

```
\defineparagraphs[chempar] [n=3,before=,after=,distance=1em]
\setupparagraphs [chempar] [1] [width=.12\textwidth]
\setupparagraphs [chempar] [2] [width=.4\textwidth]

\startchempar
Limekilns
\chempar
Hasselt has its own limekilns. These were build in 1504 and produced
quick lime up to 1956. Nowadays they are a touristic attraction.
\chempar
\chemical{CaCO_3,~,GIVES,~,CaO,~,+,~,CO_2}
\stopchempar
```

And it would come out like this:

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

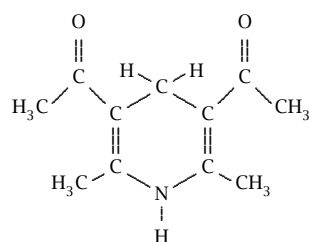
## Paragraph formatting

By the way you could also type it in a more cryptic way:

```
\chempar Limekilns
  \\ Hasselt has its own limekilns. They were build in 1504 and
  produced quick lime up to 1956. Nowadays they form a
  touristic attraction.
  \\ \chemical{CaCO_3,~,GIVES,~,CaO,~,+,~,CO_2} \\
```

The \\ are used as column separators and are essential.

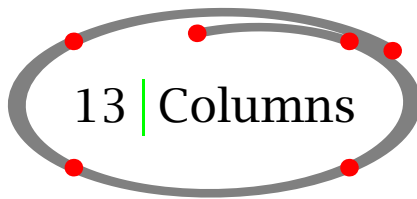
The chemical module is explained in another manual, because not everybody is interested in chemical stuff. However, chemical structures always look impressive.



Compound A

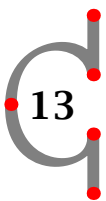
CONTEXT is relying on the PICTEX macros of M.J. Wichura to draw this kind of structures. Though the chemical module consists of only two or three commands, it takes some practice to get the right results. This is how the input looks:

```
\placeformula[-]
\startformula
\startchemical[scale=small,width=fit,top=3000,bottom=3000]
  \chemical[SIX,SB2356,DB14,Z2346,SR3,RZ3,-SR6,+SR6,-RZ6,+RZ6]
    [C,N,C,C,H,H,H]
  \chemical[PB:Z1,ONE,Z0,DIR8,Z0,SB24,DB7,Z27,PE][C,C,CH_3,0]
  \chemical[PB:Z5,ONE,Z0,DIR6,Z0,SB24,DB7,Z47,PE][C,C,H_3C,0]
  \chemical[SR24,RZ24][CH_3,H_3C]
  \bottext{Compound A}
\stopchemical
\stopformula
```



## 13 | Columns

Simple straight on texts can be displayed 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.



13

```
\startcolumns[...]=... \stopcolumns
```

```
\startcolumns[n=3,tolerance=verytolerant]
```

```
\switchtocorps[9pt]
```

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

The city has a long history since obtaining the city charter around 1252. Parts 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 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 Mastenbroek 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 Zwaterwater, the ideal surroundings for walking or cycling trips.



## Columns

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 yacht harbour 'De Molenwaard'. Sailing, fishing, swimming and canoeing can be fully enjoyed in Hasselt.

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

`\stopcolumns`

The result will be a three column text.

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

The city has a long history since obtaining the city charter around 1252. Parts 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 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 Mastenbroek 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 Zwaterwater, 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 yacht harbour 'De Molenwaard'. Sailing, fishing, swimming and canoeing can be fully enjoyed in Hasselt.

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



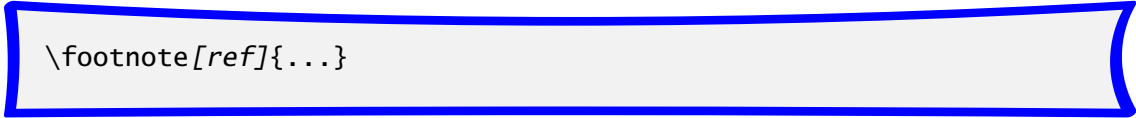
If possible a new column can be enforced with `\column`. You can set up columns with:

```
\setupcolumns[...]=...]
```



## 14 | Footnotes

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



```
\footnote[ref]{...}
```

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[ref]
```

If you have type 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.<sup>5</sup> In the Hanse period there was a lively trade in all sorts of articles such as wood, wool, metal, cloth, salt, wine and beer.<sup>5</sup> The prosperous trade caused an enormous growth of welfare in the Hanseatic towns.<sup>6</sup>

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

---

<sup>5</sup> This was the source of jealousy and fear among other towns that caused a number of wars.

<sup>6</sup> Hasselt is one of these towns.

```
\setupfootnotes[...,...=...,...]
```

## 15 | Definitions

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

```
\definedescription[...] [...,...=...,...]
```

For example:

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

It would look like this:

Hasselter juffer A sort of cookie 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 (upto 40%) mixed with herbs to give it a specific 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 third week of August.

If you've got more paragraphs in a definition you can use `\start... \stop... pair`.

```
\definedescription
  [concept]
  [place=right,
  headstyle=bold,
```

## Numbered definitions

```
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:

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 *Braderie*.

16

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

```
\setupdescriptions[...][...=...]
```

16 | Numbered definitions

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

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

For example:

```
\defineenumeration  
[remark]  
[place=top,  
text=Remark,
```

## Numbered definitions

```
inbetween=\blank,  
after=\blank]
```

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

```
\definedescription  
[concept]  
[place=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
```

Remark 1

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.

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.

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

```
\setupenumerations[...][...=...]
```

You can also vary the layout of **Remark** and **Subremark** in the example above by:

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

If a number becomes obsolete you can type:

```
\remark[-]
```

## Outlined text

In case the remark contains more than one paragraph you will have to use the command pair `\startremark ... \stopremark` that becomes available after defining **Remark** with `\defineenumeration[remark]`.

So the example above would look like this:

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

```
\startconcept{Euifeest} A harvest home to celebrate the end of a
period of hard work.
```

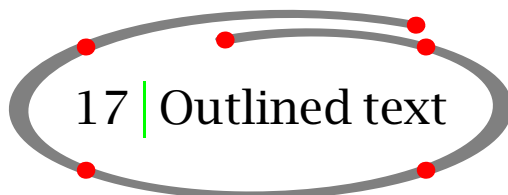
This event takes place at the end of August and lasts one week. The city is completely illuminated and the streets are decorated. This feast week ends with a `{\em Braderie}`.

```
\stopconcept
```

Remark 1

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 within the last paragraph was done with `\inframed`. This command takes care of the interline spacing.

```
\framed[height=3em,width=fit]{This box is bigger}
```

## Outlined paragraphs

This becomes:

This box is bigger

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

```
\setupcolors[state=start]
\leftlined   {\framed[width=fit]{People in Hasselt}}
\middlelined {\framed[height=1.5cm,frame=off]{have a}}
\rightlined  {\framed[background=screen]{historic background}}
```

This leads to:

People in Hasselt

have a

historic background



18

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

```
\setupframed[...]=...
```

18 | Outlined paragraphs

Complete paragraphs can be outlined with:

```
\startframedtext[...][...]=... \stopframedtext
```

```
\definefloat[intermezzo]
```

```
\placeintermezzo[here][block:bridge]{An intermezzo.}
```

```
\startframedtext[width=.8\makeupwidth]
```

It was essential for Hasselt to have a bridge across the river Zwarte

## Margin texts

Water. 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.

`\stopframedtext`

This example also illustrates the command `\definefloat`. You can find more information on this command in paragraph 37.2. The `\blank` is necessary to enforce a blank line.

It was essential for Hasselt to have a bridge across the river Zwarte Water. 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.

Intermezzo 18.1 An intermezzo.

The outlining can be set up with:

```
\setupframedtexts[...]=...]
```

19 | Margin texts

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

```
\inmargin[.1.][ref]{.2.}
```

You may remember one of the earlier examples.

`\inmarge`



```
{\externalfigure
[hass23g]
[type=tif,width=\marginwidth,frame=on]}
```



This would result in a figure in the margin. And you can imagine that it looks quite good 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)\inmarge{Street of\\ Knights} is an obvious name. In the 14th and 15th century nobility and prominent citizens lived in this street. Some of their big houses were later turned into poorhouses \inright{poor\\house}and old peoples homes. Uptill \inleft[low]{\tfd 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 broken down in 1958.

The commands \inmargin, \inleft and \inright have all 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.

The example will become:

**Street of  
Knights**

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

Uptill 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 broken down in 1958.

You can set up the margin text with:

```
\setupinmargin
```



poor  
house  
1940



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 tabel 20.1

Option	Meaning
yes	enforce a page
makeup	enforce a page without filling
no	no page
preference	prefer a new page here
bigpreference	great preference 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
reset	following commands do have effect
empty	insert an empty page
last	add pages till even number is reached
quadruple	add pages till foursome is reached

Table 20.1 Page options.

Page numbering takes place automatically but you can enforce a page number with:

```
\page[25]
```

Sometimes it is better to state a relative page number like [+2] of [-2].

The position of the page numbers on a page depend on your own preferences and if it concerns a one sided or double sided document. Page numbering can be set up with:

```
\setuppagenumbering[...]=...]
```

The preferences are placed within the brackets.

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:

```
\startpostponing ... \stoppostponing
```

```
\startpostponing
\placefigure
  {A postponed figure.}
  {\externalfigure[hass16g][frame=on,width=\textwidth]}
\page
\stoppostponing
```

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



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

```
\setupfootertexts[.1.][.2.][.3.]
```

```
\setupheadertexts[.1.][.2.][.3.]
```

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

```
\setupfootertexts[Manual][paragraph]
```

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

You can set up the head- and footline with:



Figure 20.1 A postponed figure.

## Table of contents (lists)

```
\setupheader[...][...,...=...,...]
```

```
\setupfooter[...][...,...=...,...]
```

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

```
\noheaderandfooterlines
```

## 22 | Table of contents (lists)

22

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

```
\placecontent
```

It depends on the location of this command in your document what table of contents is produced. At the top of the document it will generate a list of chapters, paragraphs etcetera. 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) paragraph titles with the corresponding paragraph numbers and page numbers.

The command `\placecontent` is available after definition with:

```
\definecombinedlist[.1.][...,2.,...][...,...=...,...]
```

## Table of contents (lists)

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

The use of this command and its conjunctive commands is illustrated for the table of contents.

```
\definelist[chapter]
\setuplist
  [chapter]
  [before=\blank,
   after=\blank,
   character=bold]

\definelist[paragraph]
\setuplist
  [paragraph]
  [alternative=d]
```

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

```
\definecombinedlist
  [content]
  [chapter, paragraph]
  [niveau=subparagraph]
```

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 itself.

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

<b>Alternative</b>	<b>Display</b>
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

Table 22.1 Alternatives for displaying lists.

Lists are set up with:

## Table of contents (lists)

```
\setuplist[...][...=...]
```

```
\setupcombinedlist[...][...=...]
```

If you want to change the layout of the generated table of contents you'll have to remember that it is a list.

```
\setupcombinedlist  
  [content]  
  [alternative=c,  
   aligntitle=nee,  
   width=2.5cm]
```

This will result in a somewhat different layout than the default one.

Lists are called up and placed with:

```
\placelist[...][...=...]
```

So if you want a table of content you type:

```
\placecontent[level=paragraph]
```

or

```
\completecontent[level=paragraph]
```

only the paragraphs will be displayed. You might need this option when you have a well structured document that has sub sub sub sub paragraphs and you don't want those in the table of contents.

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:

```
\completecontent[2.2,8.5,12.3.3]
```

A page break will occur after paragraph 2.2 and 8.5 and sub paragraph 12.3.3.

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

```
\writetolist[.1.]{.2.}{.3.}
```

```
\writebetweenlist[.1.]{.2.}
```

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

```
\section{Hotels in Hasselt}
\writetolist[paragraph]{\blank}
\writebetweenlist[paragraph]{---under construction---}
\writetolist[paragraph]{\blank}
```

## 23 | Registers

23

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. An index is sorted in alphabetical order by an auxilliary program. Sometimes the index word can not be alphabetized. For example entries such as symbols have to be stated in a redundant way 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 indexlist with:

```
\placeindex
```

or

```
\completeindex
```

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



## Synonyms

```
\defineregister[.1.][.2.]
```

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

```
\defineregister[street][streets]
```

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

```
\placestreets  
\completestreets
```

You can alter the display of the registers with:

```
\setupregister[...][...,...=...,...]
```

24

24 | Synonyms

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

```
\definesynonyms[.1.][.2.][.3.][.4.]
```

The first bracket pair contains the name of the synonym in singular the second in plural. The third bracket pair contains a command.

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

```
\definesynonyms[abbreviation][abbreviations][\infull]  
\setupsynonyms[character=kap]
```

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

## Referring to text elements

```
\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 touristic  
information on Hasselt.
```

You would obtain something like this:

The Dutch vvv (Bureau of Tourist Information) can provide you with the touristic information on Hasselt.

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

```
\input{abbrev.tex}
```

If you want to put a list of used abbreviations in your document you'll have to type:

```
\placelistofabbreviations
```

or

```
\completelistofabbreviations
```

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

The typesetting of synonyms can be influenced with:

```
\setupsynonyms
```

Look in chapter 9 where you can see another application of synonyms.

## 25 | Referring to text elements

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

```
\in{.1.}{.2.}[ref]
```

## Referring to text elements

The curly braces contain text, the brackets contain a logical label. If you have once typed a chapter header like this:

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

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, paragraphs, figures, tables, formulas etcetera.

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
```

Activities like `\in{activity}[hiking]` described on `\at{page}[hiking]` are very tiresom.

This would look like this:

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

1. swimming
2. sailing
3. hiking
4. biking

Activities like activity 3 described on page 58 are very tiresom.

You can also refer to pages. This is done with:

```
\at{.1.}{.2.}[ref]
```

For example with:

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

This command can be used in combination with:

```
\pagereference[ref]
```

and

```
\textreference[ref]{...}
```

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  
insight of the accomodations in \pagereference[accomodations]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 accomodations with `\at{page}[accomodations]`.

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

```
\placefigure  
[here]  
[fig:canals,fig:boats]  
{A characteristic picture of Hasselt.}  
{\externalfigure[hass08g][frame=on,width=5cm]}
```

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]`).

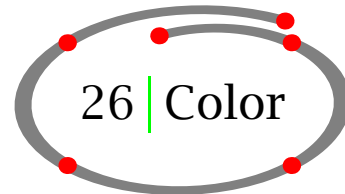
This might look like this:



Figure 25.1 A characteristic picture of Hasselt.

## Color

There are many canals in Hasselt (see figure 25.1). . . . Boats can be moored in the canals of Hasselt (see figure 25.1).



Text can be set in color.

```
\setupcolor[...]
```

The use of colors has to be activated by:

```
\setupcolor[state=start]
```

Now the basic colors are available. Basic colors are red, green and blue.

```
\startcolor[red]
```

```
Hasselt is a very \color[green]{colorful} town.
```

```
\stopcolor
```

Hasselt is a very colorful town.

On a black and white printer you will see only grey shades. In an electronic document these colors will be as defined.

You can define your own colors with:

```
\definecolor[...][...,...=...,...]
```

For example:

```
\definecolor[darkred] [r=.5,g=.0,b=.0]
```

```
\definecolor[darkgreen] [r=.0,g=.5,b=.0]
```

Now the colors darkred and darkgreen are available.



## 27 | Background of text

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

```
\startbackground ... \stopbackground
```

An example can illustrate the use:

```
\setupbackground[background=screen,corner=round]
```

```
\startbackground
```

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

```
\stopbackground
```

This would be displayed as:

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

Backgrounds can span multiple pages. With:

```
\setupbackground[...]=...]
```

you can vary the display of the backgrounds.

## 28 | Background of page areas

The background of each page area can also be set. This command looks like this:

## Alignment

```
\setupbackgrounds[.1.][...,2.,...][...,...=...,...]
```

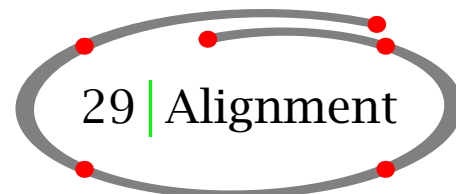
The first two bracket pairs are used to define the page areas. The last bracket pair is used for set up.

	left edge	left margin	text text	right margin	right edge
upper					
head					
text					
foot					
under					

Figure 28.1 The page areas with `\setupbackgrounds`.

If you want to have backgrounds in the gray areas of the page layout of figure 28.1 you type:

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



Horizontal and vertical alignment can be set up with:

```
\setupalign[...]
```

Single lines can be aligned with:

```
\rightlined{}
```

```
\leftlined{}
```

```
\middlelined{}
```

```
\leftlined {Hasselt was built on a sandhill.}
```

```
\middlelined {Hasselt was built on the crossing of two rivers.}
```

```
\rightlined {Hasselts name stems from hazelwood.}
```

After processing this would look like:

Hasselt was built on a sandhill.

Hasselt was built on the crossing of two rivers.

Hasselts name stems from hazelwood.

Alignment of paragraphs is done with:

```
\startalignment ... \stopalignment[...]
```

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

```
\setuptolerance[horizontal,veryrigged]
```

30 | Interactive mode in electronic documents

### 30.1 Introduction

Nowadays documents can be made electronically available for consulting on a computer and displaying on a computer screen.



Interaction means that you can click on active areas and jump to the indicated areas. 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 table of contents
- active page numbers in registers
- active page numbers, chapter numbers and figure numbers in internal references to pages, chapters, figures etcetera
- active titles, page numbers, chapter numbers in external references to other interactive documents
- active menus as navigation tools

Interactivity depends on the program you use to view the interactive document. We assume that you will use Acrobat Distiller for producing a PDF document from your PS file. It is obvious that you then will use Acrobat Reader or Acrobat Exchange for viewing.

CONTEXT 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 CONTEXT related manuals electronically (sources included) available, reverse engineering is one of the options to become more acquainted with the possibilities of CONTEXT.

## 30.2 Interactive mode

The interactive mode is activated by:

```
\setupinteraction[...,...=...,...]
```

For example:

```
\setupinteraction  
  [state=start,  
   color=green,  
   character=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 pendant because hyper links consume space. You will also notice that processing time becomes longer. Therefore it is advisable to inactivate the interactive mode as long as your document is under construction.

### 30.3 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.

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

```
\goto{.1.}{.2.}[ref]
```

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

In `\goto{Hasselt}[fig:cityplan]` all streets are build in a circular way.

In the interactive document *Hasselt* will be green and active. You will jump to a map of Hasselt.

30

### 30.4 Interaction between documents

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

```
\useexternaldocument[.1.][.2.][.3.]
```

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{...}[ref]
```

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

Look at the example below.

```
\useexternaldocument[hia][hasbook][Hasselt in August]
```

Most touristic attraction are described in `\from[hia]`. A description of the Eui||feest is found in `\from[hia::euifeest]`. A description of

the `\goto{Eui||feest}[hia::euifeest]` is found in `\from[hia]`. The `euifeest` is described on `\at{page}[hia::euifeest]` in `\from[hia]`. 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.

The double `::` indicates a reference to an external document.

After processing your input file (at least two times to get the references right) and the file `hasbook.tex` alternately you will have two PDF documents. The references above have the following meaning:

- `\from[hia]` will produce an active title you stated within the third bracket pair of `\useexternaldocument` and is linked to the first page of `hasbook.pdf`
- `\from[hia::euifeest]` will produce an active title and is linked to the page where chapter `Eui-feest` begins
- `\goto{Eui||feest}[hia::euifeest]` will produce an active word *Eui-feest* and is linked to the page where chapter `Eui-feest` begins
- `\at{page}[hia::euifeest]` will produce an active word *page* and page number and is linked to that page
- `\in{chapter}[hia::euifeest]` will produce an active word *chapter* and chapter number and is linked to that chapter

As you can see the `::` separates the (logical) file name and the destination.

### 30.5 Menus

You can define navigation tools with:

```
\defineinteractionmenu[.1.][.2.][...,...=...,...]
```

The first bracket pair is used for a logical name that can be used to recall the menu. The second pair contains the location on the screen. The third pair is used for setting up the menu.

A typical menu definition might look like this:

```
\setupcolors
  [state=start]

\setupinteraction
  [state=start,
  menu=on]

\defineinteractionmenu
  [mymenu]
```

```
[right]
[state=start,
 align=middle,
 background=screen,
 frame=on,
 width=\marginwidth,
 character=smallbold,
 color=]
```

```
\stelinteractiemenuin
[mymenu]
[{\Content[content]},
 {Index[index]},
 {\vfill},
 {Stop[ExitViewer]}]
```

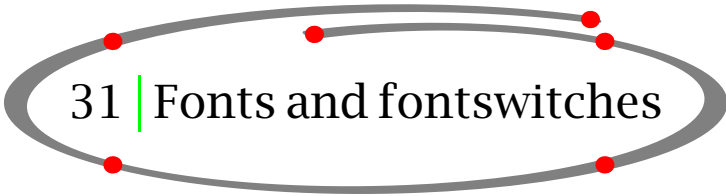
This will produce a menu on the right hand side of every screen. The menu buttons contain the text *Content*, *Index* and *Stop* with respectively the following functions: jump to the table of contents, jump to the index and leave the viewer. The labels to obvious destinations like *content* and *index* are predefined. Other predefined destinations are *FirstPage*, *LastPage*, *NextPage* and *PreviousPage*.

31

An action like *ExitViewer* 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.

Menus are set up with:

```
\setupinteractionmenu[.....][.....=.....][...,{..[ref]},...]
```



### 31.1 Introduction

The default font in `CONTEXT` is the *Computer Modern Roman* (`cmr`). You can also use

Lucida Bright (lbr) as a full alternative and symbols of the *American Society* (ams). Also a number of PostScript fonts (pos) are available.

## 31.2 Fontstyle and size

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

```
\setupcorps[...,...,...]
```

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

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

```
\switchtocorps[...,...,...]
```

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

```
\startnarrower
\switchtocorps[lbr,10pt]
\startlines
Foekepotterij, foekepotterij,
Geef mij een centje dan ga'k voorbij.
Geef mij een alfje dan blijf ik staan,
'k Zak 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.
\stoplines
\stopnarrower
```

You should notice that `\startnarrower . . . \stopnarrower` is also used as a begin and end of the fontswitch. The function of `\startlines` and `\stoplines` in this example is obvious.

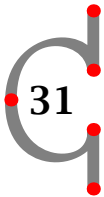
On November 10th (one day before Saint Martensday) the youth of Hasselt go from door to door to sing a special song and they accompany themselves with a *foekepot*. And they won't go away before you give them some money or sweets. The song goes like this:

Foekepotterij, foekepotterij,  
 Geef mij een centje dan ga'k voorbij.  
 Geef mij een alfje dan blijf ik staan,  
 'k Zak 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:

`\showcorps[cmr]`

[cmr]													
	<code>\tf</code>	<code>\sc</code>	<code>\sl</code>	<code>\it</code>	<code>\bf</code>	<code>\bs</code>	<code>\bi</code>	<code>\tfx</code>	<code>\tfxx</code>	<code>\tfa</code>	<code>\tfb</code>	<code>\tfc</code>	<code>\tfd</code>
<code>\rm</code>	Ag	Ag	Ag	Ag	<b>Ag</b>	<b>Ag</b>	<b>Ag</b>	Ag	Ag	Ag	Ag	Ag	<b>Ag</b>
<code>\ss</code>	Ag	Ag	Ag	Ag	<b>Ag</b>	<b>Ag</b>	<b>Ag</b>	Ag	Ag	Ag	Ag	Ag	<b>Ag</b>
<code>\tt</code>	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	Ag	<b>Ag</b>



### 31.3 Style and size switch in commands

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

`\setuphead[chapter][character=\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

### 31.4 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 available from 4pt to 12pt and is changed with `\switchtocorps`.

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 {\switchtocorps[7pt] bishops of Utrecht} did not honour these
{\slb protests}.
```

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

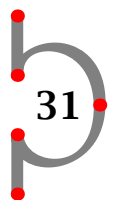
## Mintage

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*.

### 31.5 Redefining fontsize

For special purposes you can define your own fontsize.

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



A definition could look like this:

```
\definecorps[12pt][rm][tfe=cmr12 at 48pt]
{\tfe Hasselt}
```

Now `\tfe` will produce 48pt characters in the `cmr` font.

Hasselt

### 31.6 Small caps

Abbreviations like PDF (Portable Document Format) 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:

```
\kap{}
```

If you compare PDF, `\kap{PDF}` and `\sc PDF` :

PDF and PDF and PDF

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.

### 31.7 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 pilar 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.

If you walk through Hasselt you should ***watch out*** for *Amsterdammers*. An *Amsterdammer* is ***not*** a person from Amsterdam but a little stone pilar 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*.*



### 31.8 Teletype / verbatim

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

```
\starttyping ... \stoptyping
```

In the text you can use:

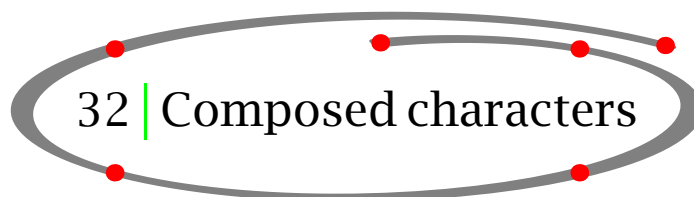
```
\type{...}
```

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

You can set up the 'typing' with:

```
\setuptyping[]
\setuptype[]
```





## 32.1 Introduction

In chapter 3 you have already seen that you have to type more than one token to obtain special characters like # \$ % & \_ { and }. Accented characters or foreign symbols have to be composed to produce the right typesetted character.

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.

## 32.2 Accents

Accented characters have to be composed in `CONTEXT`. Table 32.1 shows you how to do this. The character *u* is just an example here.

You type	You get	You type	You get
<code>\' {u}</code>	ù	<code>\u {u}</code>	ũ
<code>\' {u}</code>	ú	<code>\v {u}</code>	ǔ
<code>\^ {u}</code>	û	<code>\H {u}</code>	ǎ
<code>\" {u}</code>	ü	<code>\t {uu}</code>	ûu
<code>\~ {u}</code>	ÿ	<code>\c {u}</code>	ç
<code>\= {u}</code>	ū	<code>\d {u}</code>	ç
<code>\. {u}</code>	u	<code>\b {u}</code>	u

Table 32.1 Accents.

You don't want ì or ÿ so for an accented *i* and *j* you compose the characters as follows:

```
\{"\i}  ì
\^{\j}  ÿ
```

## 32.3 Foreign symbols

The composition of characters that appear in foreign languages is shown in table 32.2.

You type	You get	You type	You get
<code>\oe</code>	œ	<code>\O</code>	Ø
<code>\OE</code>	Œ	<code>\l</code>	ł
<code>\ae</code>	æ	<code>\L</code>	Ł
<code>\AE</code>	Æ	<code>\SS</code>	ß
<code>\aa</code>	å	<code>?‘</code>	¿
<code>\AA</code>	Å	<code>!‘</code>	¡
<code>\o</code>	ø		

Table 32.2 Foreign characters.

## 33 | Page layout

### 33

The page layout is defined by:

```
\setuplayout[...]=...]
```

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

You have to familiarize yourself with the parameters that describe your page layout. A page is divided in a number of areas such as the text, margin, head, foot etcetera. The size of these areas can be set with `\setuplayout`.

The different areas in the page layout are shown in figure 33.1.

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 (see table 33.2). This enables you to work more accurately when defining measures of columns, figures and tables. A few of these parameters are explained in table 33.1.

## Page layout

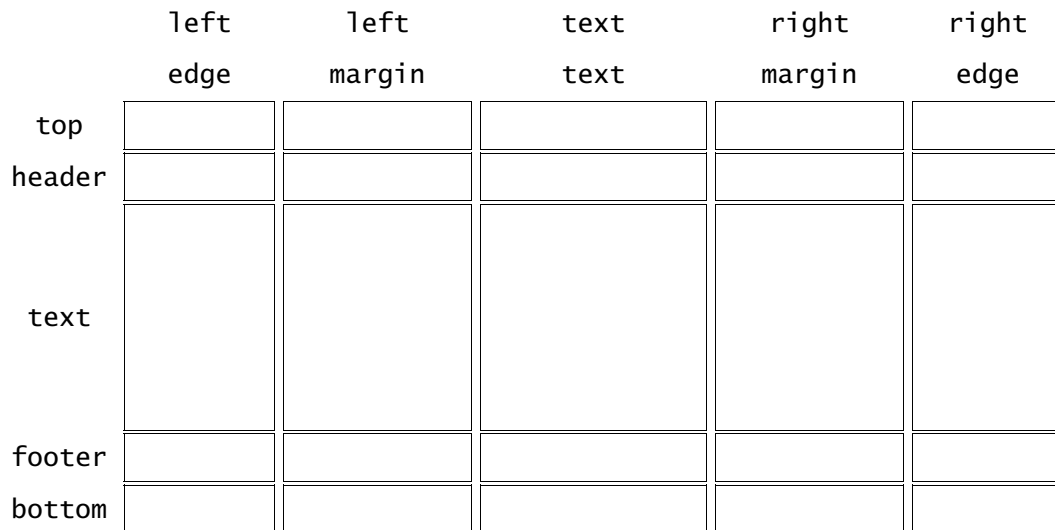


Figure 33.1 The page layout.

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 33.1 A few parameters as commands.

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

```
\placefigure
  [here]
  [fig:steppable]
  {A steppable.}
  {\externalfigure[hass19g][width=.6\textwidth,frame=on]}
```

After processing this would become:

The other distances and measures are shown in table 33.2.

The command `\setuplayout` is defined in the set up area of your input file, so before the `\starttext` command. This means that the values of the parameters have a global character and are default throughout the document. Nevertheless you might want to make slight changes in the page design for a number of pages.

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



Figure 33.2 A steppable.

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

For local changes in the page layout you can use:

```
\startlocal ... \stoplocal
```

34

For example:

```
\start  
\startlokaal  
  \setuplayout[height=+.5cm]  
\stoplokaal
```

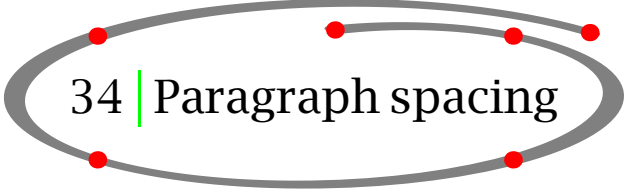
Hasselt has a completely different layout than most other cities because of its fortifications and moats.

```
\stop
```

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

Parameter	Available command
topdistance	\topheight
topheight	\topheight
headerdistance	\headerdistance
headerheight	\hoofdhoogte
topspace	\topspace
headlevel	\headerlevel
backspace	\backspace
margindistance	\margindistance
marginwidth	\marginwidth
leftmarginwidth	\leftmarginwidth
rightmarginwidth	\rightmarginwidth
edgedistance	\edgedistance
edgewidth	\edgewidth
leftedgewidth	\leftedgewidth
rightedgewidth	\rightedgewidth
paperheight	\paperheight
paperwidth	\paperwidth
makeupheight	\makeupheight
makeupwidth	\makeupwidth
textheight	\textheight
textwidth	\textwidth
footerdistance	\footerdistance
footerheight	\footerheight
bottomdistance	\bottomdistance
bottomheight	\bottomheight

Table 33.2 Parameters for page design.



## 34 | Paragraph spacing

### 34.1 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 a paragraph. You can start a new paragraph by:

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

In your  $\text{ASCII}$  input file you should use empty lines as paragraph separators. This will lead to a 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 wisdom 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.

### 34.2 Inter paragraph spacing

The vertical spacing between paragraphs can be specified by:

```
\setupwhitespace[...]
```

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

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 table like this one:

City	Area code
Hasselt	8060 - 8061

there is a small problem that must be corrected. For that purpose you could carry out a correction with:

```
\startlinecorrection ... \stoplinecorrection
```

So if you would type:

```
\startlinecorrection  
\starttable[|1|1|1|1|]  
\HL  
\NC \bf City \NC \bf Area code \NC\SR  
\HL  
\NC Hasselt \NC 8060 -- 8065 \NC\SR  
\HL  
\stoptable  
\stoplinecorrection
```

you will get a better output.

City	Area code
Hasselt	8060 - 8065

Another command to deal with vertical spacing is:

## Paragraph spacing

```
\blank[.,.,.,.,.]
```

The bracket pair is optional and within the bracket pair you can type the amount of spacing. Keywords like `small`, `middle` 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.

```
\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 suppress vertical spacing you can use:

```
\startpacked[.,.,.,.,.] ... \stoppacked
```



## Paragraph spacing

```
\defineparagraphs[city][n=2,before=,after=]
\city Hasselt (Ov) \\ Overijssel \\
\city Hasselt (Li) \\ Limburg    \\
\startpacked
\city Hasselt (Ov) \\ The Netherlands \\
\city Hasselt (Li) \\ Belgium        \\
\stoppacked
```

Hasselt (Ov)	Overijssel
Hasselt (Li)	Limburg
Hasselt (Ov)	The Netherlands
Hasselt (Li)	Belgium

It is not hard to imagine why there is also:

```
\startunpacked ... \stopunpacked
```

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

### 34.3 Indentation

If you want the paragraph to start with an indentation you can type:

```
\indenting[...]
```

in the set up area of your input file. Between the brackets you can type the preferences, default the preference is *never*.

If you choose to use indentations you will have to tell explicitly when you *do not* want to indent. This is done with:

```
\noindenting
```

You can set up the value of the indentation with:

```
\setupindenting[...]
```

## 35 | Setup commands

The set up commands are placed in the set up area of your input file. This is the area before `\starttext`. These commands have a global character and keep their values unless they are changed somewhere in mid-document.

In appendix D 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[.1.][.2.][...,...=...,...]
.1.      name
.2.      number each
style    normal bold slanted boldslanted type cap small... command
width    dimension
height   dimension
align    left right middle breedte
tolerance veryrigged rigged tolerant verytolerant
distance dimension
before   command
after    command
inner    command
command  command
line     on off
```

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

```
\setupacommand[.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.

*section*          a section name like chapter, paragraph etc.

<i>name</i>	an identifier (logical name)
<i>dimension</i>	an 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

## 36 | Defining commands / macros

CONTEXT is a set of macros based on T<sub>E</sub>X. T<sub>E</sub>X is a typographical system as well as a programming language. This means that you can do the programming yourself if you need that kind of flexibility.

You can define a new command with:

```
\define[.1.]\command{.2.}
```

Without any further explanation this command is used in the following example.

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

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

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][type=tif,width=5cm,frame=on]}}
\myputfigure{lion}{The Dutch lion is a sentry.}{hass13g}
```

Between brackets [3] indicates that you want to use three variables #1, #2 and #3. In the command call \myputfigure you have to input these variables between curly braces. The result would be:



Figure 36.1 The Dutch lion is a sentry.

Very sophisticated commands could be programmed but this is left to your own inventivity.

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

36

```
\definestartstop[...] [..., ...=..., ...]
```

For example:

```
\definestartstop
[stars]
[commandos={\inleft{\hbox to
\leftmarginwidth{\leaders\hbox{$\star$}\hfill}}},
before=\blank,
after=\blank]
```

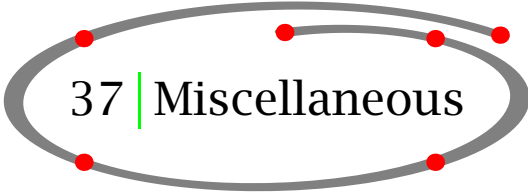
```
\startstars
{\em Hasselter Juffers} are sweet cookies but the name is no
coincidence. On July 21 in 1233 the {\em Zwartewaterklooster}
(Blackwater Monastery) was founded. The monastery was meant
```

for unmarried girls and women belonging to the nobility of Hasselt. These girls and women were called `{\em juffers}`.  
`\stopstars`

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*.

\*\*\*\*\*

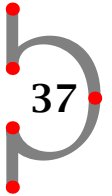


### 37.1 Floating blocks

A block in `CONTEXT` 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 10 and 11.

You can define these kind of blocks yourself with:

```
\definefloat[.1.][.2.]
```



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

```
\definefloat[intermezzo][intermezzos]
```

Now the following commands are available:

```
\placeintermezzo[] [] {} {}
\startintermezzotext ... \stopintermezzotext
\placelistofintermezzos
\completelistofintermezzos
```

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:

```
\setupcaption[...], [..., ...=..., ...]
```

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

```
\setupfloats[place=middle]
\setupcaption[place=bottom, headstyle=boldslanted]
```

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

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

```
\stopframedtext
```

37

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

Intermezzo 37.1 An intermezzo.

## 37.2 Text blocks

Another type of block is a text block. A text block for example is one or more paragraphs you want to use more than once.

You have to define a text block with:

```
\defineblock[...]
```

You can state the name of text blocks between the brackets. You can also type a list of names if you separate them by a comma. For example you can define:

```
\defineblock[dutch]
```

After defining the text block the following command is available:

```
\beginofdutch ... \endofdutch
```

Text blocks are manipulated with:

```
\hideblocks[...1,...][...2,...]
```

```
\useblocks[...1,...][...2,...]
```

```
\keepblocks[...1,...][...2,...]
```

```
\selectblocks[...1,...][...2,...][..=..]
```

An example shows the possibilities of text blocks.

```
\defineblock[dutch,english]
```

```
\hideblocks[dutch,english]
```

```
\beginofenglish[dedemsvaart-e]
```

Since 1810 the Dedemsvaart caused some prosperity in Hasselt. All ships went through the canals of Hasselt and the shops on both sides of the canals prospered.

```
\endofenglish
```

```
\beginofdutch[dedemsvaart-d]
```

Sinds 1810 veroorzaakte de Dedemsvaart enige welvaart in Hasselt. Alle schepen voeren door de grachten en de winkels aan weerszijden van de gracht floreerden.

```
\endofdutch
```

```
\useblocks[english][dedemsvaart-e]
```

If you would define these blocks consequently you could make a bilingual manual. For that purpose it is also possible to store the text blocks in an external file. This would look something like this:

```
\setupblock[dutch][file=store-d]
```

The dutch text blocks are stored in `store-d.tex` and the text fragments can be called upon by their logical names.

### 37.3 Storing text for later use

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

```
\startbuffer[...] ... \stopbuffer
```

For example:

```
\startbuffer[visit]
```

If you want to see what Hasselt has in store you come and visit it some time.

```
\stopbuffer
```

```
\getbuffer[visit]
```

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

Buffers are set up with:

```
\setupbuffer[...]=...]
```

### 37.4 Hiding text

Text can be hidden with:

```
\starthiding ... \stophiding
```

The text inbetween will not be processed.

### 37.5 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

```
\starttextrule{Hasselt -- Amsterdam}
If you draw a straight line from Hasselt to Amsterdam you would have
to cover a distance of almost 145 \Kilo \Meter.
\stoptextrule
If you draw two straight lines from Hasselt to Amsterdam you would
have to cover a distance of almost 290 \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[...,...=...,...]
```

```
\setupfillinlines[...,...=...,...]
```

These commands are introduced in the examples below:

```
\setupfillinrules[width=2cm]
\setupfillinlines[width=3cm]
\fillinrules[n=1]{\bf name}
\fillinrules[n=3]{\bf adress}
\fillinline{Can you please state the \underbar{number} of cars used
in your family.} \par
Strike any word \overstrikes{in this text}\periods[18]
```

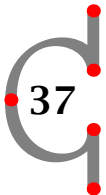
This will become:

**name** \_\_\_\_\_  
**adress** \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Can you please state the number of cars used in your family. \_\_\_\_\_

Strike any word ~~in this text~~.....

These commands are used in questionnaires. Text that is stricken or underlined can not be hyphenated.



### 37.6 Super- and subscript in text

It is very easy to put <sup>superscript</sup> and <sub>subscript</sub> in your text. How would you call this version <sup>superscript</sup><sub>subscript</sub>. It looks bad though.

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

### 37.7 Date

You can introduce the system date in your text with:

```
\currentdate
```

### 37.8 Positioning

Sometimes you feel the need to position text on a page or within a text element. You can position text with:

```
\position(.1.,.2.){.3.}
```

The parenthesis enclose the  $x, y$  coordinates, the curly braces enclose the text you want to position.

You can set up the  $x, y$  axes with:

```
\setuppositioning[...]=...]
```

You can use units and scaling factors. An example will illustrate `\position`.

```
\def\dicefive%
  {\framed
   [width=42pt,height=42pt,offset=0pt]
   {\setuppositioning
    [unit=pt,factor=12,xoffset=-11pt,yoffset=-8pt]%
    \startpositioning
     \position(1,1){$\bullet$}%
     \position(1,3){$\bullet$}%
     \position(2,2){$\bullet$}%
     \position(3,1){$\bullet$}%
     \position(3,3){$\bullet$}%
    \stoppositioning}}
\placefigure{This is five.}{\dicefive}
```

This is a rather complex example but it would look something like this.

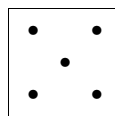


Figure 37.1 This is five.

### 37.9 Rotating text, figures and tables

In a number of cases you would like to rotate text or figures. You can rotate text and 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 '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 <sup>right</sup> to use its own seal on official documents. This seal showed Holy Stephanus known as one of the first Christian martyrs and was the <sup>patron</sup> 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 a figure just as easy:

```
\placefigure
  {The 180 \Degrees\ rotated fishinggate (de Vispoort).}
  \rotate[rotation=180]
  {\externalfigure[hass15g][frame=on]}
```

It is not always easy to understand what you get when you rotate.

### 37.10 Carriage return

A new line can be enforced with:

```
\type{...}
```

When a number of lines should be followed by *carriage return and line feed* you can use:



Figure 37.2 The 180° rotated fishinggate (de Vispoort).

```
\startlines ... \stoplines
```

```
\startlines
```

```
.  
. .  
. .
```

```
\stoplines
```

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

```
\startlines
```

```
Heimelijcken haet
```

```
eigen baet
```

```
jongen raet
```

```
Door diese drie wilt verstaen
```

```
is het Roomsche Rijck vergaen.
```

```
\stoplines
```

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

On a wooden panel in the town hall of Hasselt one 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 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.

### 37.11 Hyphenation

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

To activate a language you type:

```
\language[...]
```

Between the brackets you fill in `n1`, `fr`, `en`, `de` and `sp`.

To change from one language to another you can use the shorthand versions:

```
\n1 \en \de \fr \sp
```

If you want to know more about Hasselt you could probably best read `{\n1 \em Uit de geschiedenis van Hasselt}` by F. Peereboom.

If you want to know more about Hasselt you could probably best read *Uit de geschiedenis van Hasselt* by F. Peereboom.

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

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

### 37.12 Comment in input file

All text between `\starttext` and `\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`

## Using modules

```
% 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.

### 37.13 Input of another tex file

In a number of situations you may want to insert other  $\text{\TeX}$  files in your input file. For example when it is more efficient to specify  $\text{\CONTEXT}$  sources in more than one file in order to be able to partially process your files.

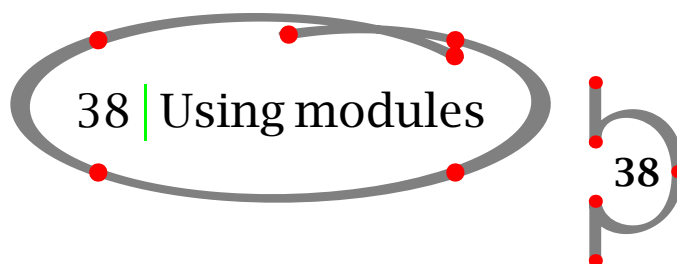
Another file (with the name `another.tex`) can be inserted by:

```
\input another.tex
```

The extension is optional so this will work too:

```
\input another
```

The command `\input` is a  $\text{\TeX}$  command.



For reasons of efficiency the author decided to implement some functionality of  $\text{\CONTEXT}$  by means of external modules. At this moment you can load the following modules:

- `chemic` for typesetting chemical structures
- `units` for using SI units
- `picTeX` for drawing pictures (is used in conjunction with module `chemic`)

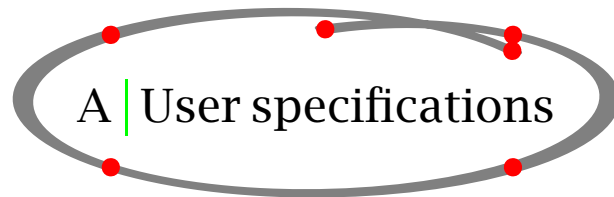
Loading is done in the set up area of your input file and done by means of:

## Using modules

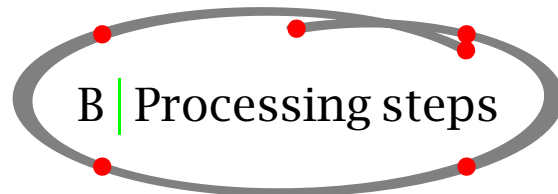
```
\usemodule[...,...,...]
```



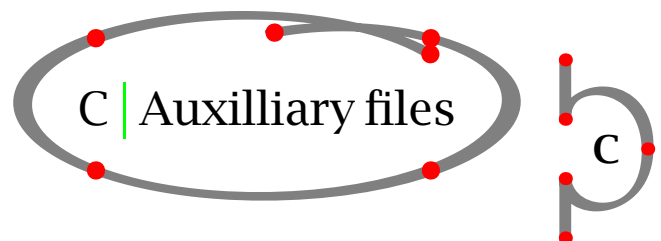
## User specifications



When `CONTEXT` is run a number of predefined parameters is loaded. These parameters are set up in the file `cont-sys.tex`. Users can define their own preferences (housestyle) in this file. Be aware of the fact that `CONTEXT` has to be able to find this file.



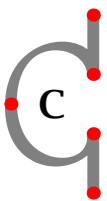
During processing `CONTEXT` writes information in the file `myfile.tui`. This information is used in the next pass. Part of this information is processed by the program `TEXUTIL`. Information on registers and lists are written in the file `myfile.tuo`. The information in this file is filtered and used (when necessary) by `CONTEXT`.




`CONTEXT` 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.

## Auxiliary files

File	Content	State
myfile.tex	text	do not remove
myfile.tui	input information	can be removed
myfile.tuo	output information	do not remove
myfile.tub	block information	do not remove
myfile.tmp	buffer information	can be removed
texutil.tuf	figure information	do not remove
myfile.dvi	typesetted text	can be removed
myfile.ps	printable text	can be removed





D | Command definitions

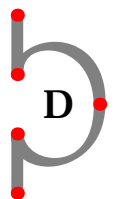
`\at{.1.}{.2.}[ref]` ◀ ▶ 58  
..1. text

`\blank[...,...]` 79  
... *n*\*small *n*\*medium *n*\*big nowhite backward white disable force reset line  
formula fixed flexible

`\chapter[ref,...]{...}` 8  
... text

`\define[.1.]\command{.2.}` 82  
..1. number

`\defineblock[...]` 86  
... name



## Command definitions

`\definecolor[...][...,...=...,...]`

60

...     *name*  
r        *text*  
g        *text*  
b        *text*  
c        *text*  
m        *text*  
y        *text*  
k        *text*

`\definecombinedlist[.1.][...,2,...][...,...=...,...]`

52

.1.       *name*  
.2.       *list*  
..=...    see p 54: `\setupcombinedlist`

`\definecorps[...,.1,...][.2.][...,...=...,...]`

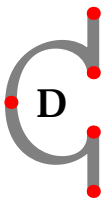
70

.1.       5pt ... 12pt  
.2.       rm ss tt mm hw cg  
tf        *file*  
bf        *file*  
sl        *file*  
it        *file*  
bs        *file*  
bi        *file*  
sc        *file*  
ex        *file*  
mi        *file*  
sy        *file*  
ma        *file*  
mb        *file*  
mc        *file*

`\definedescription[...][...,...=...,...]`

42

...        *name*  
..=...    see p 43: `\setupdescriptions`



## Command definitions

`\defineenumeration[...].1.[...].2.[...]=...]` 43

- .1. *name*
- .2. *name*
- ..=.. see p 44: `\setupenumerations`

`\definefloat[.1.][.2.]` 84

- .1. *singular name*
- .2. *plural name*

`\definehead[.1.][.2.]` 9

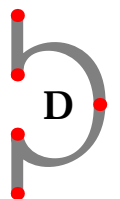
- .1. *name*
- .2. *section*

`\defineinteractionmenu[.1.][.2.][...]=...]` ◀ ▶ 66

- .1. left right top bottom *name*
- .2. left right top bottom *name*
- ..=.. see p 67: `\setupinteractionmenu`

`\defineparagraphs[...][...]=...]` 36

- ... *name*
- n *number*
- line on off
- height fit *dimension*
- before *command*
- after *command*
- inner *command*
- distance *dimension*
- tolerance veryrigged rigged tolerant verytolerant
- align left right middle



## Command definitions

`\defineregister[.1.][.2.]` 56

- .1. *singular name*
- .2. *plural name*

`\definestartstop[...][...,...=...,...]` 83

before *command*  
after *command*  
style normal bold slanted boldslanted type cap small... *command*  
commands *command*

`\definesynonyms[.1.][.2.][.3.][.4.]` 56

- .1. *singular name*
- .2. *plural name*
- .3. *command*
- .4. *command*

`\externalfigure[...][...,...=...,...]` 24

... *file*  
..=.. see p 22: `\useexternalfigure`

`\footnote[ref]{...}` 41

... *text*

`\framed[...,...=...,...]{...}` 45

..=.. see p 46: `\setupframed`  
... *text*

`\from{...}[ref]` ◀ ▶ 65

... *text*



## Command definitions

`\goto{.1.}{.2.}[ref]` ◀ ▶ 65  
.1.     *text*

`\hairline` 88

`\hideblocks[...,.1.,...][...,.2.,...]` 86  
.1.     *name*  
.2.     *name*

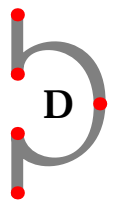
`\in{.1.}{.2.}[ref]` ◀ ▶ 57  
.1.     *text*

`\indenting[...]` 80  
...     not never no yes always

`\inmargin[.1.][ref]{.2.}` 47  
.1.     + - low  
.2.     *text*

`\keepblocks[...,.1.,...][...,.2.,...]` 86  
.1.     *name*  
.2.     all *name*

`\language[...]` 93  
...     n| fr en de sp it da



## Command definitions

`\note[ref]`

◀ ▶ 41

`\page[...]`

49

... `yes` markup no preference bigpreference left right disable last quadruple  
reset

`\pagereference[ref]`

59

`\placeformula[ref,...].1.$$2.$$`

15

.1. *text*  
.2. *formula*

`\placelist[...,...,][...,...=...,...]`

54

... *name*  
..=.. see p 54: `\setuplist`

`\position(.1.,.2.){.3.}`

90

.1. *number*  
.2. *number*  
.3. *text*

`\rotate[...,...=...,...]{...}`

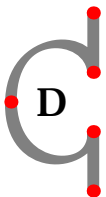
91

..=.. see p ??: `\setuprotate`

`\section[ref,...]{...}`

8

... *text*





## Command definitions

`\selectblocks[...1,...][...2,...][..=..]` 86

.1.        *name*  
.2.        *name*  
criterium    all *section*

`\setupalign[...]` 63

...        width left right middle inner outer broad height bottom line reset

`\setupbackground[.....=...]` 61

leftoffset    *dimension*  
rightoffset   *dimension*  
topoffset     *dimension*  
bottomoffset  *dimension*  
state         start stop  
..=..         see p 46: `\setupframed`

`\setupbackgrounds[.1.][...2,...][...=...]` 62

.1.        top header text footer bottom page  
.2.        leftedge leftmargin text rightmargin rechterränd  
..=..     see p 46: `\setupframed`

`\setupblank[...]` 79

...        normal standard line *dimension* big medium small fixed flexible

`\setupbuffer[.....=...]` 87

before       *command*  
after        *command*



## Command definitions

`\setupcaption[...],[...==...]` 35  
..=.. see p 24: `\setupcaptions`

`\setupcaptions[...==...]` 24

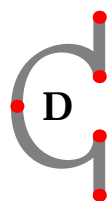
location	top <u>bottom</u> none high low middle
width	fit max <i>dimension</i>
headstyle	normal bold slanted boldslanted type cap small... <i>command</i>
style	normal bold slanted boldslanted type cap small... <i>command</i>
number	<u>yes</u> no
inbetween	<i>command</i>
align	left middle right no
conversion	<u>numbers</u> characters Characters romannumerals Romannumerals
way	<u>bytext</u> bysection

`\setupcolor[...]` 60  
... *name*

`\setupcolumns[...==...]` 40

n	<i>number</i>
line	on <u>off</u>
height	<i>dimension</i>
tolerance	veryrigged rigged <u>tolerant</u> verytolerant
distance	<i>dimension</i>
balance	<u>yes</u> no
align	yes no <u>text</u>
blank	<u>fixed line</u> flexible big medium small <u>fixed,line</u>
option	background
..=..	see p 46: <code>\setupframed</code>

`\setupcombinedlist[...][...==...]` 54  
... *name*  
level 1 2 3 4 *section*  
..=.. see p 54: `\setuplist`



## Command definitions

`\setupcorps[...,..., ...]`

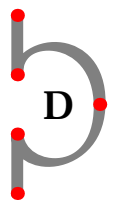
68

... *name* roman sansserif teletype handwritten calligraphy 5pt ... 12pt

`\setupdescriptions[...,..., ...][...,...=...,...]`

43

... *name*  
headstyle normal bold slanted boldslanted type cap small... *command*  
style normal bold slanted boldslanted type cap small... *command*  
color *name*  
width fit broad *dimension*  
sample *text*  
text *text*  
align left middle right  
location left right top serried inmargin inleft inright  
hang fit broad *number*  
before *command*  
inbetween *command*  
after *command*



## Command definitions

`\setupenumerations[...][...=...]` 44

...	<i>name</i>
headstyle	normal bold slanted boldslanted type cap small... <i>command</i>
headcolor	<i>name</i>
style	normal bold slanted boldslanted type cap small... <i>command</i>
color	<i>name</i>
width	fit broad <i>dimension</i>
coupling	<i>name</i>
sample	<i>text</i>
text	<i>text</i>
align	left middle right
indenting	yes no <i>dimension</i>
location	left right top serried inmargin inleft inright
hang	fit broad <i>number</i>
before	<i>command</i>
inbetween	<i>command</i>
after	<i>command</i>
levels	<i>number</i>
conversion	<u>numbers</u> characters Characters romannumerals Romannumerals
way	<u>bytext</u> bysection
blockway	<u>yes</u> no
sectionnumber	yes no
separator	<i>text</i>
stopper	<i>text</i>
number	no <i>name</i>
coupling	<i>name</i>

`\setupfillinlines[...=...]` 89

width	<i>dimension</i>
margin	<i>dimension</i>
distance	<i>dimension</i>
before	<i>command</i>
after	<i>command</i>

`\setupfillinrules[...=...]` 89

width	fit <i>dimension</i>
distance	<i>dimension</i>
before	<i>command</i>
after	<i>command</i>
style	normal bold slanted boldslanted type cap small... <i>command</i>
n	<i>number</i>
separator	<i>text</i>

## Command definitions

`\setupfloat[...][...,...=...,...]`

85

...                    *name*  
height                *dimension*  
width                 *dimension*  
pageboundaries      *list*  
..=..                 see p 46: `\setupframed`

`\setupfloats[...][...,...=...,...]`

24

location              *left right middle*  
width                 *fit dimension*  
before                *command*  
after                 *command*  
margin                *dimension*  
voorwit               *n\*small n\*medium n\*big none*  
nawit                 *n\*small n\*medium n\*big none*  
zijvoorwit            *n\*small n\*medium n\*big none*  
zijnawit              *n\*small n\*medium n\*big none*  
ntop                  *number*  
nbottom               *number*  
nlines                *number*  
..=..                 see p 46: `\setupframed`

`\setupfooter[...][...,...=...,...]`

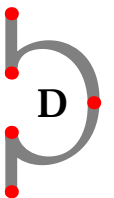
52

...                    see p 52: `\setupheader`  
..=..                 see p 52: `\setupheader`

`\setupfootertexts[.1.][.2.][.3.]`

50

.1.                    *text margin edge*  
.2.                    *text section date mark pagenummer*  
.3.                    *text section date mark pagenummer*



## Command definitions

`\setupfootnotes[...]=...`

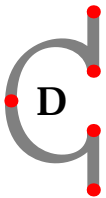
42

conversion	<u>numbers</u> characters Characters romannumerals Romannumerals
way	<u>bytext</u> <i>bysection</i>
location	<u>page</u> text
line	<u>on</u> <i>off</i>
before	<i>command</i>
after	<i>command</i>
width	<i>dimension</i>
height	<i>dimension</i>
corps	5pt ... 12pt <u>small</u> big
style	normal bold slanted boldslanted type cap small... <i>command</i>
distance	<i>dimension</i>
columndistance	<i>dimension</i>
margindistance	<i>dimension</i>
numbercommand	<code>\command#1</code>

`\setupformulas[...]=...`

17

location	left <u>right</u>
left	text
right	text



## Command definitions

<code>\setupframed[...]=...]</code>	46
<code>height</code>	fit broad <i>dimension</i>
<code>width</code>	fit broad <i>dimension</i>
<code>offset</code>	none overlay <i>dimension</i>
<code>dummy</code>	yes <u>no</u>
<code>strut</code>	<u>yes</u> no
<code>align</code>	<u>no</u> left right middle normal
<code>bottom</code>	<i>command</i>
<code>top</code>	<i>command</i>
<code>frame</code>	<u>on</u> off
<code>topframe</code>	on <u>off</u>
<code>bottomframe</code>	on <u>off</u>
<code>leftframe</code>	on <u>off</u>
<code>rightframe</code>	on <u>off</u>
<code>frameoffset</code>	<i>dimension</i>
<code>framedepth</code>	<i>dimension</i>
<code>framecorner</code>	round <u>right</u>
<code>frameradius</code>	<i>dimension</i>
<code>framecolor</code>	<i>name</i>
<code>background</code>	screen color <u>none</u>
<code>backgroundscreen</code>	<i>number</i>
<code>backgroundcolor</code>	<i>name</i>
<code>backgroundoffset</code>	<i>dimension</i>
<code>backgrounddepth</code>	<i>dimension</i>
<code>backgroundcorner</code>	round <u>right</u>
<code>backgroundradius</code>	<i>dimension</i>
<code>depth</code>	<i>dimension</i>
<code>corner</code>	round <u>right</u>
<code>radius</code>	<i>dimension</i>
<code>...</code>	<i>text</i>

<code>\setupframedtexts[...]=...]</code>	47
<code>corps</code>	5pt ... 12pt small big
<code>style</code>	normal bold slanted boldslanted type small... <i>command</i>
<code>left</code>	<i>command</i>
<code>right</code>	<i>command</i>
<code>before</code>	<i>command</i>
<code>after</code>	<i>command</i>
<code>inner</code>	<i>command</i>
<code>..=..</code>	see p 46: <code>\setupframed</code>



## Command definitions

`\setuphead[...][...,...=...,...]`

9

...	<i>section</i>
style	normal bold slanted boldslanted type cap small... <i>command</i>
textstyle	normal bold slanted boldslanted type cap small... <i>command</i>
numberstyle	normal bold slanted boldslanted type cap small... <i>command</i>
number	<u>yes</u> no
ownnumber	<u>yes</u> no
page	left right yes
continue	<u>yes</u> no
header	none empty
before	<i>command</i>
after	<i>command</i>
command	<code>\command#1#2</code>
numbercommand	<code>\command#1</code>
textcommand	<code>\command#1</code>
prefix	+ - <i>text</i>
placehead	<u>yes</u> no
incrementnumber	<u>yes</u> no

`\setupheader[...][...,...=...,...]`

52

...	<u>text</u> margin edge
state	<u>normal</u> stop start empty high none
strut	<u>yes</u> no
style	normal bold slanted boldslanted type cap small... <i>command</i>
leftstyle	normal bold slanted boldslanted type cap small... <i>command</i>
rightstyle	normal bold slanted boldslanted type cap small... <i>command</i>
leftwidth	<i>dimension</i>
rightwidth	<i>dimension</i>
before	<i>command</i>
after	<i>command</i>

`\setupheadertexts[.1.][.2.][.3.]`

50

.1.	<u>text</u> margin edge
.2.	<i>text section date mark</i> pagenumber
.3.	<i>text section date mark</i> pagenumber





## Command definitions

`\setupheads[...]=...]`

10

sectionnumber    yes no  
alternative      normal inmargin  
separator        text  
command          `\command#1#2`

`\setupindenting[...]`

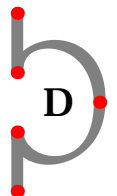
80

...      none small medium big *dimension*

`\setupinteraction[...]=...]`

◀ ▶ 64

state            start stop  
menu            on off  
page            yes no  
strut            yes no  
width            *dimension*  
height          *dimension*  
depth            *dimension*  
style            normal bold slanted boldslanted type cap small... *command*  
color            *name*  
contrastcolor   *name*  
title            *text*  
subtitle        *text*  
author          *text*  
date            *text*



## Command definitions

`\setupinteractionmenu[...][...][...=...][...,{...[ref]};▶.▶] 67`

...	left right top bottom <i>name</i>
before	<i>command</i>
after	<i>command</i>
inbetween	<i>command</i>
left	<i>command</i>
right	<i>command</i>
middle	<i>command</i>
state	<u>start</u> stop none
style	normal <u>bold</u> slanted boldslanted type cap small... <i>command</i>
color	<i>name</i>
distance	overlay <i>dimension</i>
samepage	yes empty no none
unknownreference	yes empty no none
leftoffset	<i>dimension</i>
rightoffset	<i>dimension</i>
topoffset	<i>dimension</i>
bottomoffset	<i>dimension</i>
..=..	see p 46: <code>\setupframed</code>

`\setupitemize[.1.][...2.][...=...][...=...][...=...] 14`

.1.	<i>number</i> each
.2.	<u>standard</u> <i>n</i> *broad <i>n</i> *serried packed stopper joinedup atmargin inmargin autointro
margin	<i>dimension</i> standard
width	<i>dimension</i>
distance	<i>dimension</i>
factor	<i>number</i>
items	<i>number</i>
before	<i>command</i>
inbetween	<i>command</i>
after	<i>command</i>
beforehead	<i>command</i>
afterhead	<i>command</i>
headstyle	normal bold slanted boldslanted type cap small... <i>command</i>
marstyle	normal bold slanted boldslanted type cap small... <i>command</i>
symstyle	normal bold slanted boldslanted type cap small... <i>command</i>
stopper	<i>text</i>
n	<i>number</i>
symbol	<i>number</i>

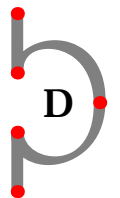


## Command definitions

`\setuplayout[...]=...]`

73

<code>width</code>	<i>dimension fit</i>
<code>height</code>	<i>dimension fit</i>
<code>backspace</code>	<i>dimension</i>
<code>topspace</code>	<i>dimension</i>
<code>margin</code>	<i>dimension</i>
<code>leftmargin</code>	<i>dimension</i>
<code>rightmargin</code>	<i>dimension</i>
<code>header</code>	<i>dimension</i>
<code>footer</code>	<i>dimension</i>
<code>top</code>	<i>dimension</i>
<code>bottom</code>	<i>dimension</i>
<code>leftedge</code>	<i>dimension</i>
<code>rightedge</code>	<i>dimension</i>
<code>headerdistance</code>	<i>dimension</i>
<code>footerdistance</code>	<i>dimension</i>
<code>topdistance</code>	<i>dimension</i>
<code>bottomdistance</code>	<i>dimension</i>
<code>leftmargindistance</code>	<i>dimension</i>
<code>rightmargindistance</code>	<i>dimension</i>
<code>leftedgedistance</code>	<i>dimension</i>
<code>rightedgedistance</code>	<i>dimension</i>
<code>backoffset</code>	<i>dimension</i>
<code>topoffset</code>	<i>dimension</i>
<code>style</code>	normal bold slanted boldslanted type cap small... <i>command</i>
<code>mark</code>	on <u>off</u>
<code>location</code>	middle <u>singlesided</u> doublesided



\setuplist[...][...][...]=...]

54

...	<i>name</i>
state	<u>start</u> stop
alternative	a <u>b</u> c d none
coupling	on <u>off</u>
criterium	<u>section</u> <u>local</u> previous all
pageboundaries	<i>list</i>
style	normal bold slanted boldslanted type cap small... <i>command</i>
numberstyle	normal bold slanted boldslanted type cap small... <i>command</i>
textstyle	normal bold slanted boldslanted type cap small... <i>command</i>
pagestyle	normal bold slanted boldslanted type cap small... <i>command</i>
color	<i>name</i>
command	\command#1#2#3
numbercommand	\command#1
textcommand	\command#1
pagecommand	\command#1
interaction	<u>sectionnumber</u> text pagenumber all
before	<i>command</i>
after	<i>command</i>
left	<i>text</i>
right	<i>text</i>
label	yes <u>no</u>
prefix	<u>yes</u> no
pagenumber	<u>yes</u> no
sectionnumber	yes no
aligntitle	yes <u>no</u>
margin	<i>dimension</i>
width	<i>dimension</i>
height	<i>dimension</i> fit broad
depth	<i>dimension</i> fit broad
offset	<i>dimension</i>
distance	<i>dimension</i>
separator	<i>text</i>
symbol	none 1 2 3 ...
expansion	yes <u>no</u> command



## Command definitions

`\setuppagenumbering[...]=...` 49

alternative	<u>singlesided</u> <u>doublesided</u>
location	header footer left right <u>middle</u> margin marginedge inleft inright
conversion	<u>numbers</u> characters Characters romannumerals Romannumerals
style	normal bold slanted boldslanted type cap small... <i>command</i>
left	<i>text</i>
right	<i>text</i>
way	bytext bysection <u>bypart</u>
text	<i>text</i>
numberseparator	<i>text</i>
textseparator	<i>text</i>
sectionnumber	yes no
separator	<i>text</i>
state	<u>start</u> stop
command	<code>\command#1</code>

`\setupparagraphs[.1.][.2.][...]=...` 36

.1.	<i>name</i>
.2.	<i>number</i> each
style	normal bold slanted boldslanted type cap small... <i>command</i>
width	<i>dimension</i>
height	<i>dimension</i>
align	left right middle <u>breedte</u>
tolerance	veryrigged rigged <u>tolerant</u> verytolerant
distance	<i>dimension</i>
before	<i>command</i>
after	<i>command</i>
inner	<i>command</i>
command	<i>command</i>
line	on <u>off</u>

`\setuppositioning[...]=...` 90

unit	<u>cm</u> pt em mm ex sp in
factor	<i>number</i>
scale	<i>number</i>
offset	<u>yes</u> no
xstep	<u>absolute</u> relative
ystep	<u>absolute</u> relative



## Command definitions

`\setupregister[...][...,...=...,...]` 56

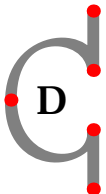
...	<i>singular name</i>
n	<i>number</i>
balance	yes <u>no</u>
align	yes <u>no</u>
style	normal bold slanted boldslanted type cap small... <i>command</i>
indicator	<u>yes</u> no
coupling	yes <u>no</u>
sectionnumber	yes no
criterion	<i>section</i> local all <u>part</u>
distance	<i>dimension</i>
symbol	1 2 ... n a ...
interaction	<u>pagenumber</u> text
expansion	yes <u>no</u>
referencing	<u>on</u> off
command	<code>\command#1</code>

`\setupsectionblock[...][...,...=...,...]` 7

...	<i>name</i>
number	<u>yes</u> no
page	yes <u>right</u>
before	<i>command</i>
after	<i>command</i>

`\setuptables[...]=...,...]` 32

distance	small <u>medium</u> big
corps	5pt ... 12pt small big
HL	<i>number</i> small medium big none
VL	<i>number</i> small medium big none
commands	<i>command</i>
background	screen color <u>none</u>
backgroundscreen	<i>number</i>
backgroundcolor	<i>name</i>



## Command definitions

`\setupthinrules[...=...]` 88

distance    *small medium big*  
n            *number*  
before      *command*  
after       *command*  
height      *dimension*  
depth       *dimension*

`\setupwhitespace[...]` 78

...        *none small medium big line fixed fix *dimension**

`\startalignment ... \stopalignment[...]` 63

..=...    see p 63: `\setupalign`

`\startbackground ... \stopbackground` 61

`\startbuffer[...]` ... `\stopbuffer` 87

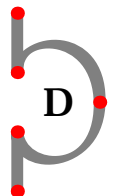
...        *name*

`\startcolumns[...=...]` ... `\stopcolumns` 39

..=...    see p 40: `\setupcolumns`

`\startcombination[...]` ... `\stopcombination` 23

...        *n\*m*



## Command definitions

`\startformula ... \stopformula` 15  
... *formula*

`\startframedtext[...][...=...=...]` ... `\stopframedtext` 46  
... left right middle  
..=.. see p 47: `\setupframedtexts`

`\starthiding ... \stophiding` 87

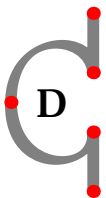
`\startitemize[.....][.....=.....]` ... `\stopitemize` 10  
... a A KA n N m r R KR *number* continue standard *n*\*broad *n*\*serried packed stopper  
joinedup atmargin inmargin intro columns  
..=.. see p 14: `\setupitemize`

`\startlegend[.1.]\leg.2.\.3.\.4.\\leg` ... `\stoplegend` 18  
.1. two  
.2. text  
.3. text  
.4. text

`\startlinecorrection ... \stoplinecorrection` 78

`\startlines ... \stoplines` 92

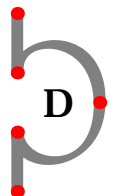
`\startlocal ... \stoplocal` 75





## Command definitions

<code>\startpacked[...,...] ... \stoppacked</code> ... blank	79
<code>\startpostponing ... \stoppostponing</code>	50
<code>\starttable[...] ... \stoptable</code> ... <i>text name</i>	26
<code>\starttyping ... \stoptyping</code>	71
<code>\startunpacked ... \stopunpacked</code>	80
<code>\subject[ref,...]{...}</code> ... <i>text</i>	8
<code>\subsection[ref,...]{...}</code> ... <i>text</i>	8
<code>\subsubject[ref,...]{...}</code> ... <i>text</i>	9
<code>\switchtocorps[...,...]</code> ... 5pt ... 12pt small big global	68



## Command definitions

`\textreference[ref]{...}` 59  
... text

`\thinrule` 88

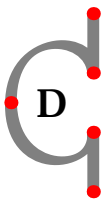
`\thinrules[..=..]` 88  
..=.. see p 88: `\setupthinrules`

`\title[ref,...]{...}` 8  
... text

`\type{...}` 71  
... text

`\useblocks[...1,...][...2,...]` 86  
.1. name  
.2. name

`\useexternaldocument[.1][.2][.3]` ◀ ▶ 65  
.1. name  
.2. file  
.3. text



## Command definitions

`\useexternalfigure[.1.][.2.][...,...=...,...]` 22

.1. *name*  
.2. *file*  
scale *number*  
factor *max fit broad*  
wfactor *number max broad fit*  
hfactor *number max broad fit*  
width *dimension*  
height *dimension*  
frame *on off*  
preset *yes no*  
preview *yes no*

`\usemodule[...,..., ...]` 95

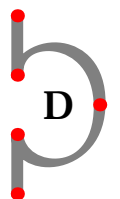
... *name*

`\writebetweenlist[.1.]{.2.}` 55

.1. *section name*

`\writetolist[.1.]{.2.}{.3.}` 54

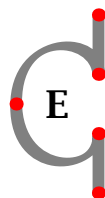
.1. *section name*





# E | Command Index

<code>\abbreviation</code>	56	<code>\em</code>	71
<code>\adaplayout</code>	73	<code>\en</code>	93
<code>\at</code>	57, 65	<code>\externalfigure</code>	20
<code>\bf</code>	69	<code>\fact</code>	17
<code>\blank</code>	77	<code>\footnote</code>	41
<code>\chapter</code>	8	<code>\FR</code>	25
<code>\color</code>	60	<code>\fr</code>	93
<code>\column</code>	39	<code>\framed</code>	45
<code>\completecontent</code>	52	<code>\from</code>	65
<code>\completeindex</code>	55	<code>\getbuffer</code>	87
<code>\completelistofabbreviations</code>	56	<code>\goto</code>	65
<code>\completeregister</code>	55	<code>\hairline</code>	87
<code>\crlf</code>	91	<code>\head</code>	10
<code>\currentdate</code>	89	<code>\hideblocks</code>	85
<code>\DC</code>	25	<code>\high</code>	89
<code>\de</code>	93	<code>\HL</code>	25
<code>\defineblock</code>	85	<code>\in</code>	57, 65
<code>\definecolor</code>	60	<code>\indenting</code>	80
<code>\definecombinedlist</code>	52	<code>\index</code>	55
<code>\definecorps</code>	70	<code>\inframed</code>	45
<code>\definedescription</code>	42	<code>\infull</code>	56
<code>\defineenumeration</code>	43	<code>\inleft</code>	47
<code>\definefloat</code>	84	<code>\inmargin</code>	47
<code>\definelist</code>	52	<code>\input</code>	94
<code>\defineparagraphs</code>	36	<code>\inright</code>	47
<code>\defineregister</code>	55	<code>\item</code>	10
<code>\definesymbol</code>	10	<code>\kap</code>	70
<code>\definesynonyms</code>	56	<code>\language</code>	93
<code>\DL</code>	25	<code>\leftlines</code>	62
<code>\DR</code>	25		



## Command Index

`\leg` 17  
`\loadabbreviations` 56  
`\lohi` 89  
`\LOW` 25  
`\low` 89  
`\LR` 25

`\margintitle` 47  
`\middlelined` 62  
`\MR` 25

`\NC` 25  
`\n1` 93  
`\noheaderandfooterlines` 50  
`\noindenting` 80  
`\nowhitespace` 77  
`\NR` 25

`\page` 48  
`\pagereference` 57  
`\par` 77  
`\paragraph` 8, 77  
`\placecontent` 52  
`\placefigure` 20  
`\placeformula` 15  
`\placeindex` 55  
`\placeintermezzo` 84  
`\placelistofabbreviations` 56  
`\placeregister` 55  
`\placetable` 25  
`\position` 90  
`\procent` 18  
`\promille` 18

`\rightlined` 62  
`\rm` 69  
`\rotate` 90

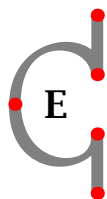
`\setupalign` 62  
`\setupbackground` 61  
`\setupbackgrounds` 61

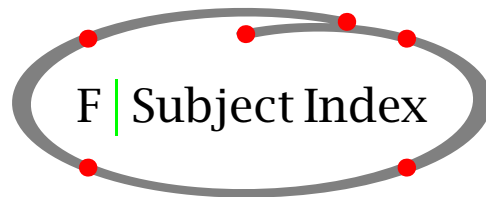
`\setupblank` 77  
`\setupblock` 85  
`\setupbuffer` 87  
`\setupcaptions` 20, 25, 84  
`\setupcolors` 60  
`\setupcolumns` 39  
`\setupcombinedlist` 52  
`\setupcorps` 68  
`\setupdescriptions` 42  
`\setupenumerations` 43  
`\setupfigures` 20  
`\setupfloat` 84  
`\setupfloats` 20, 25, 84  
`\setupfooter` 50  
`\setupfootertexts` 50  
`\setupfootnotes` 41  
`\setupformulas` 15  
`\setupframed` 45  
`\setupframedtext` 46  
`\setuphead` 8  
`\setupheader` 50  
`\setupheadertexts` 50  
`\setupheads` 8  
`\setupindenting` 80  
`\setupinteraction` 64  
`\setupitemize` 10  
`\setuplayout` 73  
`\setuplinecorrection` 77  
`\setuplist` 52  
`\setuppositioning` 90  
`\setuppagenumbering` 48  
`\setupparagraphs` 36  
`\setupregister` 55  
`\setupsynonyms` 56  
`\setuptables` 25  
`\setupthinrules` 87  
`\setuptolerance` 62  
`\setuptype` 71  
`\setuptyping` 71  
`\setupwhitespace` 77  
`\showlayout` 73



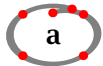
## Command Index

<code>\showmakeup</code>	73	<code>\subsubject</code>	8
<code>\showsetups</code>	73	<code>\switchtocorps</code>	68
<code>\sl</code>	69	<code>\tfa</code>	69
<code>\sp</code>	93	<code>\tfb</code>	69
<code>\SR</code>	25	<code>\tfc</code>	69
<code>\ss</code>	69	<code>\tfd</code>	69
<code>\startbackground</code>	61	<code>\thinrule</code>	87
<code>\startbuffer</code>	87	<code>\thinrules</code>	87
<code>\startchemical</code>	36	<code>\THREE</code>	25
<code>\startcolumns</code>	39	<code>\title</code>	8
<code>\startcombination</code>	20, 25	<code>\tt</code>	69
<code>\startfact</code>	17	<code>\TWO</code>	25
<code>\startfiguretext</code>	20	<code>\type</code>	71
<code>\startformula</code>	15	<code>\typebuffer</code>	87
<code>\startframedtext</code>	46	<code>\unit</code>	18, 56
<code>\starthiding</code>	87	<code>\useblocks</code>	85
<code>\startitemize</code>	10	<code>\useexternaldocument</code>	65
<code>\startlegend</code>	17	<code>\useexternalfigures</code>	20
<code>\startlines</code>	91	<code>\VL</code>	25
<code>\startlocal</code>	73	<code>\whitespace</code>	77
<code>\startpacked</code>	77	<code>\writebetweenlist</code>	52
<code>\startpostponing</code>	48	<code>\writetolist</code>	52
<code>\starttable</code>	25		
<code>\starttextrule</code>	87		
<code>\starttyping</code>	71		
<code>\startunpacked</code>	77		
<code>\subject</code>	8		
<code>\subparagraph</code>	8		



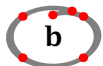


F | Subject Index



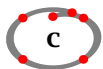
a

accents 72  
alignment 62  
AMS 67  
auxilliary files 96



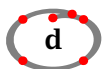
b

background  
page areas 61  
text 61



c

carriage return 91  
color 60  
columns 36, 39  
columns in itemize 10  
comment 93  
Computer Modern Roman 67  
cont-sys.tex 96



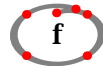
d

date 89  
definition 42  
Acrobat Distiller 63  
dvi-file 5



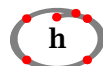
e

emphasized 71  
Acrobat Exchange 63



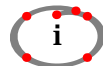
f

figure 20  
floating blocks 20, 25, 84  
fontsize 70  
font  
size 68  
style 68  
footnote 41  
foreign symbols 72  
formula 15




h

headers 8  
hiding text 87  
hyphenation 93



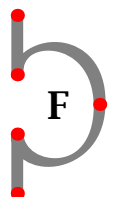
i

indentation 80  
input file 3  
processing 5  
input other  $\TeX$ -files 94  
interaction  
external 65  
internal 65  
interactive mode 64  
inter paragraph spacing 77  
itemize 10



l

label 57  
language 93



## Subject Index

layout 73, 81  
legends 17  
lines 87  
list 52  
Lucida Bright 67

**m**

margin text 47

**n**

numbered definition 43

**o**

outline

paragraph 46  
text 45

**p**

page areas 61  
page breaking 48  
page design 73  
pagefooter 50  
page header 50  
page numbering 48  
paragraph 77  
indentation 80  
% in input file 93  
positioning 90  
ps-fonts 67

**r**

Acrobat Reader 63

referring 57  
register 55  
rotating 90

**s**

set up 81  
SI-unit 18  
small caps 70  
special characters 6  
storing text 87  
subscript 89  
superscript 89  
synonyms 56

**t**

table of contents 52  
tables 25  
text blocks 85  
TEXUTIL 96  
tub-file 96  
tuf-file 96  
tui-file 96  
tuo-file 96, 96  
type 71

**u**

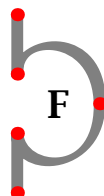
units 18

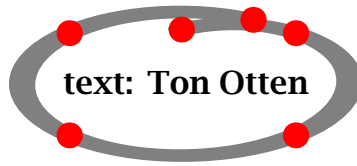
**v**

verbatim 71

**F**



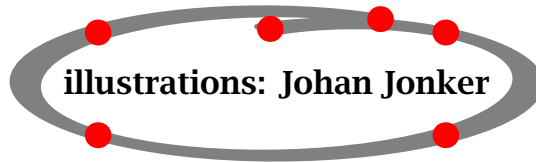


A gray oval with a thick border, containing six red dots arranged in a circle around the text.

**text: Ton Otten**

A gray oval with a thick border, containing six red dots arranged in a circle around the text.

**design: Hans Hagen**

A gray oval with a thick border, containing six red dots arranged in a circle around the text.

**illustrations: Johan Jonker**

A gray oval with a thick border, containing six red dots arranged in a circle around the text.

**PRAGMA  
Ridderstraat 27  
8061GH Hasselt NL  
+31 (0)38 477 53 69  
pragma@pi.net**

A gray oval with a thick border, containing six red dots arranged in a circle around the text.

**1997 October 28**