An Extension of the \LaTeX\ theorem environment

Frank Mittelbach

Abstract
The macros described in this paper yield an extension of the \LaTeX\ theorem mechanism. It is designed to satisfy the different requirements of various journals. Thus, the layout of the "theorems" can be manipulated by determining a "style". This article describes not only the use, but also the definition, of the necessary macros.

Preface to version 2.1
This version is identical to 2.0g described in TUGboat 10\#3 except for some internal defaults which are now set depending on the used font selection scheme.
This was done to avoid unpleasant surprises if the new font selection scheme is in force. For further details see section 3 and [2, 3].

1 Introduction
For our purposes here, "theorems" are labelled enunciations, often set off from the main text by extra space and a font change. Theorems, corollaries, conjectures, definitions, and remarks are all instances of "theorems". The "header" of these structures is composed of a label (such as \textsc{Theorem} or \textsc{Remark}) and a number which serializes an item in the sequence of items with the same label.
Shortly after the introduction of \LaTeX\ at the Fachbereich Mathematik in Mainz, the desire to manipulate the layout of "theorems" arose. In Mainz, the following two conventions came into general use:
1. The number of the theorem is shown in the margin.
2. There is a line break at the end of the theorem header.
Additionally, some journals require different formats which depend on the "sort of theorem": e.g. often remarks and definitions are set in \textsc{rm}, while \textsc{it} is employed for main theorems.
Confronted with these requirements, a theorem environment was developed in Mainz which allows separate determination of the layout of the "theorems sets", comparable to \textsc{pagestyle}.

2 The user interface
2.1 Defining new theorem sets
\texttt{\newtheorem} As in the original \LaTeX\ version, the command \texttt{\newtheorem} defines a new "theorem set" or "theorem-like structure". Two required arguments name the new environment and give the text to be typeset with each instance of the new "set", while an optional argument determines how the "set" is enumerated:
\texttt{\newtheorem{foo}{bar}} The theorem set \texttt{foo} (whose name is \texttt{bar}) uses its own counter.
\texttt{\newtheorem{foo2}{foo2\texttt{}}{bar2}} The theorem set \texttt{foo2} (printed name \texttt{bar2}) uses the same counter as the theorem set \texttt{foo}.
\texttt{\newtheorem{foo3}{bar3}[section]} The theorem set \texttt{foo3} (printed name \texttt{bar3}) is enumerated within the counter section, i.e. with every new \texttt{section} the enumeration begins again with 1, and the enumeration is composed from the section-number and the theorem counter itself.

\footnote{This file has version number v2.1b, last revised 90/06/04, documentation dated 90/06/04.}
Additionally, the command \texttt{\texttheoremsyle} can define the layout of various, or all, theorem sets. It should be noted that any theorem set defined by \texttt{\newtheorem} is typeset in the \texttt{\texttheoremsyle} that is current at the time of the definition. Thus, the following

\begin{verbatim}
\texttt{\texttheoremsyle\{break\} \newtheorem\{Cor\}\{Corollary\}}
\texttt{\texttheoremsyle\{plain\} \newtheorem\{Exa\}\{Example\}\{section\}}
\end{verbatim}

leads to the result that the set Cor is formatted in the style break, while the set Exa and all the following ones are formatted in the style plain, unless another \texttt{\texttheoremsyle} follows. Since the definitions installed by \texttt{\newtheorem} are global, one also can limit \texttt{\texttheoremsyle} locally by grouping braces.

The choice of the font for the theorem body is completely independent of the chosen \texttt{\texttheoremsyle}; this has proven to be very advantageous. For example,

\begin{verbatim}
{\texttt{\texttheorembodyfont\{\rm\}} \newtheorem\{Rem\}\{Remark\}}
\end{verbatim}

defines a theorem set Rem, which will be set in \texttt{\rm} in the current layout (which in our example is plain). As with \texttt{\texttheoremsyle}, the \texttt{\texttheorembodyfont} chosen is that current at the time of \texttt{\newtheorem}. If \texttt{\texttheorembodyfont} is not specified or one defines \texttt{\texttheorembodyfont\{}\texttt{\}}, then the font used will be that defined by the \texttt{\texttheoremsyle}.

It is also possible to customize the font used for the theorem headers. This is, however, a global declaration, and therefore there should be at most one \texttt{\texttheoremsyle} declaration in the preamble.\footnote{If it is actually necessary to have different header fonts, one has to define new \texttt{\texttheoremsyle} sets (substituting the desired font) or specify the information directly in the \texttt{\newtheorem} declaration (the unclean variant).}

Two additional parameters affect the vertical space around the theorem environments: \texttt{\texttheorem preskipamount} and \texttt{\texttheorem postskipamount} define, respectively, the spacing before and after such an environment. These parameters apply for all theorem sets and can be manipulated with the ordinary length macros. They are rubber lengths, ('skips'), and therefore can contain plus and minus parts.

Since the definition of theorem sets should—most sensibly—be placed in the preamble, we only allow installation there. It is therefore possible to release the memory used here after \texttt{\begin\{document\}}, in order to make room for other applications.

2.2 Existing theorem styles

The following theorem styles exist to date:

\begin{verbatim}
plain \hspace{1cm} This theorem style emulates the original \LaTeX{} definition, except that additionally the parameters \texttt{\texttheoremsyle\{\}} are used.
break \hspace{1cm} In this style, the theorem header is followed by a line break.
marginbreak \hspace{1cm} The theorem number is set in the margin, and there is a line break as in break.
changebreak \hspace{1cm} Like break, but with header number and text interchanged.
change \hspace{1cm} Header number and text are interchanged, without a line break.
margin \hspace{1cm} The number is set in the left margin, without a line break.
\end{verbatim}

All styles (except plain) select \texttt{\small} as the default \texttt{\texttheorembodyfont}.

2.3 Examples

Given the above theorem sets Cor, Exa and Rem, suppose that the preamble also contains the declarations:

\begin{verbatim}
\texttt{\texttheoremsyle\{marginbreak\} \newtheorem\{Lem\}\{Lemma\}}
\end{verbatim}
Then the following are some typical examples of the typeset output resulting from their use.

**Corollary 1**
*This is a sentence typeset in the theorem environment Cor.*

**Example 2.1** *This is a sentence typeset in the theorem environment Exa.*

**Remark 1** *This is a sentence typeset in the theorem environment Rem.*

2 **Lemma (Ben User)**
*This is a sentence typeset in the theorem environment Lem.*

3 **Definition (Very Impressive Definition)** *This is a sentence typeset in the theorem environment Def.*

The last two examples show the effect of the optional argument to a theorem environment (it is the text typeset in parentheses).

### 3 Special Considerations

Theoremheader and body are implemented as a unit. This means that the \texttt{\theorembodyfont} will inherit characteristics of the \texttt{\theorembodyfont} if the new font selection scheme [3] is in force. Thus, if for example \texttt{\theorembodyfont} is \texttt{it} and \texttt{\theorembodyfont} is \texttt{bf} the font selected for the header will have the characteristics ‘bold extended italic’. If this is not desired one should set the \texttt{\theorembodyfont} to something like

\texttt{\theorembodyfont\{normalshape\bf\}}

i.e. supplying all necessary font informations explicitly.

### 4 Acknowledgements

The publication of this set of macros was only possible with the help of Christina Busse (translating the manuscript into English), Joachim Pense (playing the rôle of typist), Chris Rowley (looking everything over) and many others providing useful suggestions.

### 5 Definition of the Macros

If the file has been loaded before, we abort immediately. If not, the current version of the style is shown on the screen and in the transcript file.

1 \texttt{\iffundef\{theoremstyle\}}\{\endinput\}
2 \texttt{\typeout\{Style option: 'theorem' \fileversion \space \space}
3 \texttt{\space \space \\<\filedate\> (FMi)}
4 \texttt{\typeout\{English documentation as of \space \space \space}
5 \texttt{\space \space \<\doctype\> (FMi)}

#### 5.1 Definition of theorem styles and fonts

All the definitions in this file are done globally to allow inputting this file inside a group.
\theoremstyle

Before a theorem style can be installed, the chosen style must be known. For that reason, we must test to see that \texttt{\th@style} is known or, more precisely, that it is different from \relax. If the style is not known then \texttt{\th@plain} is used.

6 \edef\theoremstyle#1\%
7 \ifundefined{\th@#1}\{\warning
8 \{Unknown theoremstyle \texttt{\#1}'. Using \texttt{\plain}\%
9 \theoremstyle\{\plain\}\%

We save the theorem style to be used in the token register \texttt{\theorem@style}.

10 \{\theorem@style\{\#1\}\%

Now we “evaluate” the theorem style: this means, we call the macro \texttt{\th@style} which will activate the relevant definitions which are contained in a separate file. This is done in a group to suppress changes to the current font. This could otherwise pose problems together with the new font selection scheme\footnote{When I printed the original article using the new font selection scheme I ended with a document with slanted typefaces (text headings and all) simply because one of the theorem styles used \texttt{\sc} at toplevel.} if the \texttt{\th@style} is evaluated a second time.

11 \begingroup
12 \sename \texttt{\th@\the\theorem@style} \endsename
13 \endgroup

\begin{theorem}
\end{theorem}
\opargbegintheorem

We reset \texttt{\begin{theorem}} and \texttt{\opargbegintheorem} to \texttt{\relax} since these commands are no longer necessary at toplevel. This will save a few tokens.

14 \global\let\begin{theorem}\relax
15 \global\let\opargbegintheorem\relax

\theorem@style

Obviously the token register used above has to be allocated. To assure the utmost compatibility with the original \LaTeX{} definition, we set the default theorem style to \texttt{\plain}, which implements the usual \LaTeX{} convention.

16 \newtoks\theorem@style
17 \global\theorem@style\{\plain\}

\theorembodyfont

For the theorem font, we simply use a token register, whose contents can be inserted into the definition of the theorem set.

18 \newtoks\theorembodyfont
19 \global\theorembodyfont\{\}

\theoremheaderfont

The font for the theorem headers is handled differently because this definition applies to all theorem styles.

20 \edef\theoremheaderfont#1\{\edef\theorem@headerfont[#1]\%

After using the macro once it is redefined to produce an error message.

21 \edef\theoremheaderfont##1\%
22 \texttt{\space should be used \texttt{only once.}}\}

\theoremheaderfont

To set the \texttt{\theorem@headerfont} default we first test if the new font selection scheme is in force.

24 \ifx\normalshape\undefined

If not we define it to expand into \texttt{\bf}. We don’t use \texttt{\let} just in case a following style option redefines this macro.

25 \edef\theorem@headerfont\{\bf\}

\end{document}
Otherwise we reset the current shape before calling \bf.

\begin{verbatim}
\th@plain
\th@break
\th@marginbreak
\th@changebreak
\th@change
\th@margin
\end{verbatim}

The different styles are defined in macros such as \th@plain. Since memory space is precious in “non-Big-versions”, we have to avoid offering too many unused definitions. Therefore we define these styles in separate files that can be loaded on demand. Thus the commands themselves only load these files.

\begin{verbatim}
27 \def\th@plain{\input thp.sty}
28 \def\th@break{\input thb.sty}
29 \def\th@marginbreak{\input thmb.sty}
30 \def\th@changebreak{\input thch.sty}
31 \def\th@change{\input thc.sty}
32 \def\th@margin{\input thm.sty}
\end{verbatim}

This list will be expanded when new styles become available. For testing, just append new theorem substyles as document options.

5.2 Definition of a new theorem set

As already pointed out, a new theorem environment can be defined in three different ways:

\begin{verbatim}
\newtheorem{Lem}{Lemma}
\newtheorem{Lem}{Lemma}[section]
\newtheorem{Lem}{Theorem}[Lemma]
\end{verbatim}

The function of the macro \newtheorem is to recognize these cases and then to branch into one of the three macros \@nthm, \@nthm or \@nthm. This mechanism is adopted unchanged from [1]; the essential point here is that, for example, in the second case, the arguments Lem, Lemma and section are passed over to the macro \@nthm.

We inspect this case first because the others present fewer problems, and thus are easily derived from this one.

\begin{verbatim}
\@nthm
\end{verbatim}

For our example arguments, the macro \@nthm must fulfill the following:

- Define a new \texttt{\LaTeX} counter ‘Lem’
- reset this counter within a \texttt{section}
- define the macro \texttt{\thLem}
- define the environment macros \texttt{\Lem} and \texttt{\endLem} using the current \texttt{\theoremnestyle}
  and \texttt{\theoremmstyle}.

Obviously, all this should happen only if the first argument of \@nthm (i.e. Lem in our example) is chosen so as not to conflict with any previously defined commands or environments. This test is performed by the \texttt{\LaTeX} macro \texttt{\@ifdefinable}.

\begin{verbatim}
33 \if@definable\@nthm\@1\@3\expandafter\@ifdefinable\csname #1\endcsname
\end{verbatim}

Therefore, the first argument of \@ifdefinable is the expansion (in the example, \texttt{\Lem}) of \texttt{\csname #1\endcsname}. The second argument is executed only if the test has been completed successfully.

% Now we define the new counter. The names of the \texttt{\LaTeX} macros employed should speak for themselves:

\begin{verbatim}
35 \definecounter{#1}\addtoreset{#1}{#3}
\end{verbatim}

In defining \texttt{\thLem} we must generate the desired macro name by use of \expandafter and \texttt{\csname}.

\begin{verbatim}
36 \expandafter\xdef\csname the\#1\endcsname
\end{verbatim}
An `\xdef` is used in order to make the definition global, and to ensure that it contains
the replacement texts of `\@thmcountersp` and `\@thmcounter`. However, not every-
thing should be expanded. For example, it saves space to use `\thesection` instead of its—at times—lengthy expansion.

```latex
\expandafter \noexpand \csname the#3\endcsname
\@thmcountersp \@thmcounter{#1}\%
```

Thus with the defaults of \LaTeX, `\th@Lem` would be replaced by the command sequence
`\thesection \arabic{\lem}`.

We will now look at the current definition of the macro which is executed at the beginning of
the actual environment (in our example this macro is `\lem`). It should be noted that we use an
`\expandafter \expandafter` to expand only certain parts of the replacement text
at the time of the definition.

```latex
\def \@tempa{\@global \@namedef{#1}\%}
\expandafter \@tempa \expandafter \expandafter\%
```

First, the macro that contains the current definitions of `\@begintheorem` and
`\@par@theorem` should be called up. The name of this macro—as is already known—
has the form `th@\{theorem style\}`; therefore, it must be called by

```latex
\csname th@\the \ theorem@style\endcsname
```

In addition the default theorem font should be changeable, i.e. we have to insert the
contents of `\the\bodyfont` for that reason, we expand even further, beyond `\endcsname`, and thus insert the contents of the token register `\the\bodyfont` in the
replacement text.

```latex
\expandafter \endcsname \the \ bodyfont
```

Now it is time to call the macro `\@thm` which takes over the further processing. It has
two arguments: the current counter name (in our example, `\lem`), and the text of the
label (in our example, `\Lemma`).

```latex
\@thm{#1}{#2}\%
```

With this, the 'sub-definition' is complete. The macro `\end@theorem` ends a theorem
environment and is, so far, nothing but an `\end@list`. (Hence it is defined globally,
and not within the theorem styles.) Therefore, we can set it equivalent to the macro
that ends the theorem set (in our example, `\end@lem`). However, if some day theorem
styles exist that do change `\end@theorem`, we would have to use the commented-out
line instead.

```latex
\expandafter \begin@theorem \csname end@#1\endcsname \@end@\the@\end@theorem\%
```

With these commands all the required definitions are employed, unless the test
`\@ifdefinable` has failed. Therefore, we end the second argument of this macro
and with it the definition of `\@thm`.

```latex
\end@theorem
```

`\syn@thm` The definition of `\syn@thm` is completely analogous. In this case the new counter that is
defined is not reset within another counter; thus the definition of `\the ...` is simplified:

```latex
\gdef\syn@thm@#2{\expandafter \@ifdefinable \csname #1\endcsname
\let \csname end@#1\endcsname \syn@thm@\%
```

```latex
\expandafter \syn@thm@\expandafter \syn@thm@\expandafter
\expandafter \syn@thm@\expandafter \syn@thm@\%
```

3 These two macros can be defined by the document style. Their default values produce a ',' as separation and an arabic representation of the number.

4 This has to be changed as soon as theorem styles that change `\@end@theorem` exist. In such a case, all existing styles must be changed as well since they will have to reset the macro.
The rest of the definition corresponds literally to that of `\xnthm`:

```
\def@tempa{\global\namedef{#3}\expandafter \@tempa
\expandafter{\csname th@the \the@theoremstyle \expandafter }
\@thm
\% \global\namedef{end\#1}\@endtheoremm
\global \expandafter \let \csname end\#1\endcsname \csname \endcsname}
```

All other parts of the definition can be adopted from `\xnthm`. We have to remember, though, that in this case the name of the current counter and the theorem label have moved to the second and third arguments.

```
\def@tempa{\global\namedef{#3}\expandafter \@tempa
\expandafter{\csname th@the \the@theoremstyle \expandafter }
\@thm
\% \global\namedef{end\#1}\@endtheorem
\global \expandafter \let \csname end\#1\endcsname \csname \endcsname}
```

### 5.3 Macros that are employed in a theorem environment

`\@thm` has to increase the current counter. Then, depending on whether the environment has (or does not have) an optional argument, it has to branch into either `\beginthm` or `\opargthm`.

```
\def@thm{#2}{\restepcounter{#2}}
```

Now we start a trivlist environment, and give `\topsep` and `\topsepadd` the values of the skip registers \texttt{\thm@skipamount} and \texttt{\thm@postskipamount}. The value in \texttt{\topsep} is the vertical space that is inserted by the first (and only) \texttt{\item} in our trivlist whilst \texttt{\topsepadd} is inserted by \texttt{\endparem} at the end of that trivlist environment. By using these registers, we obtain the desired space around a theorem environment.

```
\trivlist
\@topsep \thm@skipamount \% used by first \item
\@topsepadd \thm@postskipamount \% used by \endparem
```

Now we have to test whether an optional argument has been given.

```
\if@fnextchar \%
```

If there is an optional argument, we will call `\@thm`, and move the arguments read back into the input stream.

```
{\@thm[#1][#2]}
```

If not, we call `\beginthm`. Its first argument is the name of the theorem set (hence the second argument of `\thm`). Its second argument is the macro that produces the current number.

```
{\beginthm[#2]{\csname the\#1\endcsname}\ignorespaces}
```

Both these macros were originally called by `\thm`. We do not need `\xthm` anymore, hence we reset it to \relax. The definition of `\relax` has not changed at all from its definition in \TeX. In order to make the macros easier to understand, we will nevertheless present it (commented out).

```
\global \let \xthm \relax
```
The primitive \ignorespaces in \@thm and \@thm is needed to remove the spaces between the \begin{...} and the actual text.

5.4 Definition of the theorem styles

As already pointed out, the theorem styles, defined below, are only loaded when necessary. Note that all these styles, except plain, have \sl as the default body font.

5.4.1 The plain style

As the following macros use \empty, we have to locally set the \catcode of this symbol to "letter". This happens within a group, so that we do not have to worry about which \catcode that symbol had before.

\begin{group}
\setlength{\baselineskip}{30pt}
\item \@beginplain
\item \@opargbegintheorem
\def\@beginplain{
\it
Then we define \@beginplain and \@opargbegintheorem. These two macros define how the header of a theorem is typeset. \@opargbegintheorem will be called if a theorem environment with an optional argument is encountered; otherwise, the header is constructed by calling \@beginplain. If one of these macros is executed, we are within a trivial environment started by \@thm. So the theorem header is produced with the \item command.

Instead of specifying the header font directly, all standard theorem styles use the \@theoremmheaderfont macro to allow customization. The extra space (\labelsep) is necessary because of problems in the trivial environment.

\def\@opargbegintheorem##1##2[\@arg]{
\item[\skiplabelsep \@theoremmheaderfont {##1} \ ##2]}

The definition of \@opargbegintheorem is completely analogous. The only difference is the fact that there exists a third argument (which is the optional parameter of the environment and contains additional information about the theorem). Customarily we enclose it in parentheses.

\def\@opargbegintheorem##1##2##3[\@arg]{
\item[\skiplabelsep \@theoremmheaderfont {##1} \ ##2 \ (#3)]}
We conclude with an \endgroup to restore the \catcode of \empty.
\endgroup

5.4.2 The break style

This style option is stored in the file thb.sty. For the next two lines see the documentation for \th@plain on page 8.
\begin{group}
\makeatletter
\ifdefined\theorem@style\{\input{theorem.sty}\}\}
\end{group}

First, we show the version\(^6\) of this file:
\begin{verbatim}
\type{Style-Option: \textbackslash {theorembreak} \textbackslash fileversion \textbackslash space\space}
\textbackslash {\textbackslash filedate} \textbackslash (FMi)\}
\type{English Dokumentation \textbackslash {\spaces}{\spaces}{\spaces}\textbackslash space\space}
\textbackslash {\textbackslash space} \textbackslash {\textbackslash docdate} \textbackslash (FMi)\}
\end{verbatim}

The \textbackslash {theorembreak} produces a line break after the name of the theorem. The font is \textbackslash {s}. Hence, we define \textbackslash {theorembreak} as follows:
\begin{verbatim}
\def\theorembreak{\s}
\edef\@begintheorem##1##2{\item[\%}
\rlap{\vbox{\hbox{\hskip \labelsep \theorembheaderfont \#1 \ ##2}}]
\hbox{\strut}}\}%
\end{verbatim}

We run into the following problem: it is not possible to create the header with \item[\textbackslash {title}] and then start a new line by, for example, \textbackslash {\hbox}\. Such a definition will fail whenever a list environment follows immediately. With the above construction, the \textbackslash {\hbox} causes the switch \textbackslash infalse (cf. definition of \textbackslash list and \textbackslash {\textbackslash trivlist in [1]} to be set to false and so the following list will insert additional vertical space \textbackslash {\topskip}. This is quite annoying. Therefore, we create the line break within the \item. In order to ensure that the text will begin at the proper position in the following line, we simply pretend that the label does not take any room.\(^7\)
\begin{verbatim}
\rlap{\vbox{\hbox{\hskip \labelsep \theorembheaderfont \#1 \ ##2}}%}
\hbox{\strut}}\}%
\end{verbatim}

Again, the definition of \textbackslash @opargbegintheorem is completely analogous.
\begin{verbatim}
\def\@opargbegintheorem##1##2##3{%
\item[\rlap{\vbox{\hbox{\hskip \labelsep \theorembheaderfont
\rlap{\hbox{\strut}}\]%}
\end{verbatim}

5.4.3 The changebreak style

This style option is stored in the file thcb.sty.
\begin{group}
\makeatletter
\ifdefined\theorem@style\{\input{theorem.sty}\}\}
\end{group}

First, we show the version\(^8\) of this file:
\begin{verbatim}
\type{Style option: \textbackslash {theorembbreak} \textbackslash fileversion \textbackslash space\space}
\textbackslash {\textbackslash filedate} \textbackslash (FMi)\}
\type{English Dokumentation \textbackslash {\spaces}{\spaces}{\spaces}\textbackslash space\space}
\textbackslash {\textbackslash space} \textbackslash {\textbackslash docdate} \textbackslash (FMi)\}
\end{verbatim}

\(^6\) This file has version number v2.1b, last revised 89/10/28, documentation dated 89/02/15.

\(^7\) This will lead to problems whenever very high symbols occurring in the line tower into the heading. So, something else has to be done here sometime.

\(^8\) This file has version number v2.1b, last revised 89/10/28, documentation dated 89/09/18.
\section*{\texttt{\textbackslash th@changebreak}}

The change-break theorem style is like \texttt{break} but with interchange of theorem name and theorem number. Thus we define \texttt{\textbackslash th@changebreak} as follows:

\begin{verbatim}
\def\th@changebreak{\sl
\def\@begintheorem#1#2{\item
  \vspace{2\baselineskip}
  \hspace{1em}
  \noindent\texttt{\textbackslash theorem@headerfont} #2 \ #1
  \texttt{\textbackslash strut}\}
\def\@opargbegintheorem#1#2{\item
  \vspace{2\baselineskip}
  \hspace{1em}
  \noindent\texttt{\textbackslash theorem@headerfont} #2 \ #1
  \texttt{\textbackslash strut}\}
\endgroup
\end{verbatim}

\subsection*{5.4.4 The change style}

This style option is stored in the file \texttt{thc.sty}.

\begin{verbatim}
\begin{group}
  \makeatletter
  \ifdefined\theoremstyle{\input{theorems.sty}}\{}
  \begin{verbatim}
  \typeout{\textit{Style option: \texttt{theorems-change}.} \texttt{fileversion}\space\space
  \texttt{\textbackslash dates} (FMi)}
  \typeout{\textit{English documentation \texttt{\@spaces}\texttt{\@spaces}\texttt{\@spaces}}\space\space
  \texttt{\textbackslash dates} (FMi)}
\end{verbatim}
\end{verbatim}
\end{group}
\end{verbatim}

\section*{\texttt{\textbackslash th@change}}

The change theorem style corresponds to the change break style without a linebreak after the header. To say it in another way, it's the same as the plain style but with number and name interchanged and \texttt{\sl} as the default font.

\begin{verbatim}
\def\th@change{\sl
\def\@begintheorem#1#2{\item
  \vspace{2\baselineskip}
  \hspace{1em}
  \noindent\texttt{\textbackslash theorem@headerfont} #2 \ #1
  \texttt{\textbackslash strut}\}
\def\@opargbegintheorem#1#2{\item
  \vspace{2\baselineskip}
  \hspace{1em}
  \noindent\texttt{\textbackslash theorem@headerfont} #2 \ #1 \ (#3)}
\endgroup
\end{verbatim}

\subsection*{5.4.5 The marginbreak style}

This style option is the one used most often at Mainz. It is saved in the file \texttt{thmb.sty}.

\begin{verbatim}
\begin{group}
  \makeatletter
  \ifdefined\theoremstyle{\input{theorems.sty}}\{}
  \begin{verbatim}
  \typeout{\textit{Style option: \texttt{theorems-margin-break}.} \texttt{fileversion}\space\space
  \texttt{\textbackslash dates} (FMi)}
  \typeout{\textit{English documentation \texttt{\@spaces}\texttt{\@spaces}\texttt{\@spaces}\texttt{\@spaces}}\space\space
  \texttt{\textbackslash dates} (FMi)}
\end{verbatim}
\end{verbatim}
\end{group}
\end{verbatim}

\section*{\texttt{\textbackslash th@marginbreak}}

The margin break style is nearly the same as the change break style. The only difference is the placement of the theorem number. We use \texttt{\textbackslash lap} to place it in the left margin.

In this style \texttt{\textbackslash labelsep} denotes the separation between the number and the text.

\begin{verbatim}
\def\th@marginbreak{\sl
\def\@begintheorem#1#2{\item
  \vspace{2\baselineskip}
  \hspace{1em}
  \noindent\texttt{\textbackslash theorem@headerfont} #2 \ #1
  \texttt{\textbackslash strut}\}
\endgroup
\end{verbatim}

\footnote{This file has version number v1.1b, last revised 89/10/28, documentation dated 89/09/18.}

\footnote{This file has version number v2.1c, last revised 90/02/17, documentation dated 90/02/17.}
5.4.6 The margin style

This style option is stored in the file thm.sty.

\begin{group} \makeatletter
\ifundefined{theorem@style}{\input{theorem.sty}}{}

First, we show the version\textsuperscript{11} of this file:

\begin{verbatim}
\typeout{Style option: `theorem-margin' \fileversion \space \space
  \<\filedate> (PMi)}
\typeout{English documentation \@spaces\@spaces\@spaces\space \space
  \<\space \space \<\docdate> (PMi)}

\th@margin
\end{group}

Again this is only a variant of the theorem styles described above without any new
ideas.

\begin{verbatim}
\def\th@margin{sl}
\def@begintheorem#12\item
  [\theorem@headerfont \hspace{\labelsep #1}]\%
\def@opargbegintheorem\hspace{\labelsep #3}\%
\item[\theorem@headerfont \hspace{\labelsep #1} (#3)]\%
\end{group}

5.5 Final Definitions

\th@em prest@kipamount
\th@em post@kipamount

The skip parameters that regulate the vertical empty space before and after the
\begin{document} \begin{theorem} \end{theorem} \end{document}
environment have to be allocated as well.

\newskip\th@em prest@kipamount
\newskip\th@em post@kipamount

Since we have used the same values for all theorem sets, we now can assign them.

\begin{verbatim}
\global\setlength\th@em prest@kipamount{12\pt plus 5\pt minus 3\pt}
\global\setlength\th@em post@kipamount{8\pt plus 3\pt minus 1.5\pt}
\end{verbatim}

@end theorem

The same holds for the macro \@@end@trivlist, which ends a \begin{document} \begin{trivlist} \end{trivlist} \end{document}
environment. Since it is the same for all theorem sets, it is removed from the macros \th@t{style}.
It simply ends the \begin{trivlist} \end{trivlist} environment, which was begun in \@thm.

\begin{verbatim}
\global\let\end@thm=\end@trivlist
\end{verbatim}

@end@preamble@cmd

All macros defined above are to be used only in the preamble. Therefore, we insert
them in \@preamble@cmd (separated by the macro \@do). All the commands not allowed after the
@document are stored here. This is achieved by having @document (with the help of \@do) redefine all macros in \@preamble@cmd to a \LaTeX error routine call.

\begin{verbatim}
{\def@do@noexpand\do@noexpand}
\def@preamble@cmd{\@preamble@cmd \do@noexpand \@thm \do@nthm \do@thm}
\do@new@theorem \do@theorem@style \do@theorem@headerfont
\end{verbatim}

\textsuperscript{11} This file has version number v2.1b, last revised 89/10/28, documentation dated 89/09/18.
Finally we declare the \texttt{plain} theorem style to be the default.

\begin{verbatim}
\theoremstyle{plain}
\end{verbatim}

\section*{References}


Frank Mittelbach
Electronic Data Systems
(Deutschland) GmbH
Eisenstraße 56
D-6090 Rüsselsheim
Federal Republic of Germany
Bitnet: \texttt{pzf5xz@drueds2}