

Using the `amsthm` package

Richard Kaye

12th August 1998

1 Introduction

The `amsLATEX` `amsthm` package extends `LATEX`'s `\newtheorem` command (see 'Cross references in `LATEX`' and the file `crossref.tex`), and is very useful. The following is taken from the documentation.

The `amsthm` package is loosely derived from `theorem.sty` version 2.1c; it adds `\newtheorem*` for unnumbered environments and changes the extensibility support for loading extra theorem styles from external files: now it is done through the package option mechanism.

Here are some examples showing the kinds of theorem environment declarations that are possible with the `amsthm` package.

```
\newtheorem{prop}{Proposition}
\newtheorem{thm}{Theorem}[section]
\newtheorem{lem}[thm]{Lemma}
\newtheorem*{Zorn}{Zorn's Lemma}
```

```
\theoremstyle{definition}
\newtheorem{dfn}{Definition}
```

```
\theoremstyle{remark}
\newtheorem*{rmk}{Remark}
```

The first four statements all define environments using the default theorem style ('plain'), since there is no prefatory `\theoremstyle` declaration. The first statement defines an automatically numbered `prop` environment whose headings will look like this: Proposition 1, Proposition 2, and so forth. The second statement defines an environment `thm` with numbers subordinate to section numbers, so the headings will look like this: Theorem 1.1, Theorem 1.2, Theorem 1.3, ..., (in section 2:) Theorem 2.1, Theorem 2.2, and so forth. The third statement defines a `lem` environment whose numbers will interleave in sequence with the theorem numbers: Theorem 1.3, Lemma 1.4, Lemma 1.5, Theorem 1.6, and so forth. The fourth statement defines a special unnumbered lemma named 'Zorn's Lemma'. The remaining two `\newtheorem` statements have no special features except for the `\theoremstyle` declarations that cause the `dfn` and `rmk` environments to have some differences in appearance.

There are three basic styles provided: The ‘plain’ style produces bold headings and italic body text; the ‘definition’ style produces bold headings and normal body text; the ‘remark’ style produces italic headings and normal body text.

The examples here give the following.

Proposition 1. *This is a proposition.*

Theorem 1.1. *This is a theorem.*

Lemma 1.2. *This is a lemma.*

Zorn’s Lemma. *This is Zorn’s lemma.*

Definition 1. This is a definition.

Remark. This is a remark.

I suggest you use these commands as in the examples given, using whatever ‘theorem style’ you prefer for theorems, lemmas, remarks, examples, definitions and so on. (You don’t *have* to use the theorem style ‘definition’ for definitions only!) Personally, I prefer to use the ‘definition’ style and ‘remark’ style for everything, and not use the ‘plain’ style at all.

The `amsthm` package also has an environment for proofs, which you use as follows.

```
\begin{proof} We prove Lemma~5 by first considering the
value of  $x$ . \ldots
\end{proof}
```

giving

Proof. We prove Lemma 5 by first considering the value of x □

For more advanced L^AT_EXing, there are some more features added to the `amsthm` package. You most likely will never need them, but in case you do, here is the documentation.

A `\swapnumbers` command allows theorem numbers to be swapped to the front of the theorem headings. Putting `\swapnumbers` in your document preamble will cause *all following* `\newtheorem` statements to produce number-first headings. (To provide maximum control, `\swapnumbers` is designed so that it can be used more than once; each time it is used, theorem numbers will be swapped to the opposite side for all following `\newtheorem` statements. But rarely will it need to be invoked more than once per document.)

There is a `\newtheoremstyle` command provided to make the creation of custom theoremstyles fairly easy.

Usage:

```
                #1
\newtheoremstyle{NAME}%
                #2          #3          #4
                {ABOVESPACE}{BELOWSPACE}{BODYFONT}%
```

```

#5      #6      #7      #8
{INDENT}{HEADFONT}{HEADPUNCT}{HEADSPACE}%
#9
{CUSTOM-HEAD-SPEC}

```

Leaving the ‘indent’ argument empty is equivalent to entering `Opt`. The ‘headpunct’ and ‘headspace’ arguments are for the punctuation and horizontal space between the theorem head and the following text. There are two special values that may be used for ‘headspace’: a single space means that a normal interword space should be used; “`\newline`” means that there should be a line break after the head instead of horizontal space. The ‘custom-head-spec’ argument follows a special convention: it is interpreted as the replacement text for an internal three-argument function `\thmhead`, i.e., as if you were defining

```
\renewcommand{\thmhead}[3]{...#1...#2...#3...}
```

but omitting the initial `\renewcommand{\thmhead}[3]`. The three arguments that will be supplied to `\thmhead` are the name, number, and optional note components. Within the replacement text you can (and normally will want to) use other special functions `\thmname`, `\thmnumber`, and `\thmnote`. These will print their argument if and only if the corresponding argument of `\thmhead` is nonempty. For example

```
{\thmname{#1}\thmnumber{ #2}\thmnote{ (#3)}}
```

This would cause the theorem note `#3` to be printed with a preceding space and enclosing parentheses, if it is present, and if it is absent, the space and parentheses will be omitted because they are inside the argument of `\thmnote`.

Finally, if you have an extra bit of arbitrary code that you want to slip in somewhere, the best place to do it is in the ‘body font’ argument.

The `\newtheoremstyle` command is designed to provide, through a relatively simple interface, control over the style aspects that are most commonly changed. Clearly it cannot serve for all possible theorem styles. Therefore there is a second interface provided to allow arbitrary theorem styles by reading suitable definitions from a separate file whose name ends with `.thm`. If the desired style is far from any of the basic styles provided by the `amsthm` package, writing the definitions may require some expertise in \LaTeX ’s macro language.

Suppose that you wanted to make a theorem style ‘exercise’ for exercises. Create a file called `exercise.thm` and in it define a command `th@exercise` following the form of the commands `th@plain`, `th@definition`, `th@remark` given below. Then to use the new style, write

```

\usepackage[exercise]{amsthm}
...
\theoremstyle{exercise}

```

Similarly, it's possible to place a group of related `\newtheoremstyle` statements in a `.thm` file, let's say `stygroup.thm`, so that they could be loaded on demand in various documents by

```
\usepackage[stygroup]{amsthm}
```

This strategy fails if you want to load a `.thm` file in a document preamble and the `amsthm` package has already been loaded in the documentclass (e.g., `amsart`). Then you need to use a statement such as

```
\PassOptionsToPackage{stygroup}{amsthm}
```

before the `\documentclass` command.