WIKIBOOKS

LaTeX/Labels and Cross-referencing

Contents

Introduction

In LaTeX, you can easily reference almost anything that can be numbered, and have LaTeX automatically updating the numbering for you whenever necessary. The objects which can be referenced include chapters, sections, subsections, footnotes, theorems, equations, figures and tables[1]. The commands to be used do not depend on what you are referencing, and they are:

\label{*marker*}
    Used to give the object you want to reference a *marker* — a name which can be used to refer to that object later.
\ref{*marker*}
    Used to reference an object with the specified marker. This will print the number that was assigned to the object.
\pageref{*marker*}
    Used to print the page number where the object with the specified marker is found.

LaTeX will calculate the right numbering for the objects in the document; the *marker* you have used to label the object will not be shown anywhere in the document. Instead, LaTeX will replace the string "\ref{*marker*}" with the right number that was assigned to the object. If you reference a *marker* that does not exist, the compilation of the document will be successful but LaTeX will return a warning:

```
LaTeX Warning: There were undefined references.
```

and it will replace "\ref{*unknown-marker*}" with "??" — so that it will be easier to find in the document.

As you may have noticed, this way of cross-referencing is a two-step process: first the compiler has to store the labels with the right number to be used for referencing, then it has to replace the \ref with the right number.

Because of that, you would have to compile your document twice to see the output with the proper numbering. If you only compile it once, then LaTeX will use the older information collected in previous compilations (which might be outdated), and the compiler will inform you by printing the following message at the end of the compilation:

```
LaTeX Warning: Label(s) may have changed. Rerun to get cross-references
right.
```

Using the command \pageref{} you can help the reader to find the referenced object by providing also the page number where it can be found. You could write something like:

```
See figure~\ref{fig:test} on page~\pageref{fig:test}.
```

Since you can use exactly the same commands to reference almost anything, you might get a bit confused after you have introduced a lot of references. It is common practice among LaTeX users to add a few letters to the label to describe *what* you are referencing. Some packages, such as fancyref, rely on this meta information. Here is an example:

ch:	chapter
sec:	section
subsec:	subsection
fig:	figure
tab:	table
eq:	equation
lst:	code listing
itm:	enumerated list item
alg:	algorithm
app:	appendix subsection

Following this convention, the label of a figure will look like \label{fig:*my_figure*}, etc. You are not obligated to use these prefixes, and can in fact use any string as an argument of \label{...}, but these prefixes can become increasingly useful as your document grows in size.

Another suggestion: try to avoid using numbers within labels. You are better off describing *what* the object is about. This way, if you change the order of the objects, you will not have to rename all your labels and their references.

If you want to be able to see the markers you are using in the output document as well, you can use the `showkeys` package; this can become very useful as you develop your document. For more information see the Packages section.

Examples

Here are some practical examples, but you will notice that they are all the same because they all use the same commands.

Sections

```
\section{Greetings}
\label{sec:greetings}

Hello!

\section{Referencing}

I greeted in section~\ref{sec:greetings}.
```

1 Greetings

Hello!

2 Referencing

I greeted in section 1.

You could place the label anywhere in the section; however, in order to avoid confusion, it is better to place it immediately after the *beginning* of the section. Note how the marker starts with *sec:*, as suggested before. The label is then referenced in a different section, where the tilde (~) indicates a non-breaking space.

Pictures

You can reference a picture by inserting it in the `figure` floating environment.

```
\begin{figure}
  \centering
    \includegraphics[width=0.5\textwidth]
{gull}
  \caption{Close-up of a gull}
  \label{fig:gull}
\end{figure}
Figure~\ref{fig:gull} shows a photograph of a
gull.
```


Figure 1: Close-up of a gull

Figure 1 shows a photograph of a gull.

When a label is declared within a float environment, the `\ref{...}` will return the respective figure/table number, but it must occur *after* the caption. When declared outside, it will give the section number. To be completely safe, the label for any picture or table can go within the

`\caption{}` command, as follows:

```
\caption{Close-up of a gull\label{fig:gull}}
```

For more, see the Floats, Figures and Captions section about the `figure` and related environments.

Fixing wrong labels

The command `\label` must appear after (or inside) `\caption`. Otherwise, it will pick up the current section or the list number instead of what is intended.

```
\begin{figure}
  \centering
  \includegraphics[width=0.5\textwidth]{gull}
  \caption{Close-up of a gull} \label{fig:gull}
\end{figure}
```

Issues with links to tables and figures handled by hyperref

In case you use the package `hyperref` to create a PDF, the link to a table or a figure will point to its caption instead, which is always below the table or the figure itself[2]. As a result, the table or the figure will not be visible if it is above the pointer, which means that some scrolling-up would be required. If you want the link to point to the top of the image, you can give the option `hypcap` to the `caption` package[3]:

```
\usepackage{caption} % hypcap is true by default so [hypcap=true] is optional in \usepackage[hypcap=true]{caption}
```

Formulae

Here is an example showing how to reference formulae:

```
\begin{equation} \label{eq:solve}
x^2 - 5 x + 6 = 0
\end{equation}

\begin{equation}
x_1 = \frac{5 + \sqrt{25 - 4 \times
6}}{2} = 3
\end{equation}

\begin{equation}
x_2 = \frac{5 - \sqrt{25 - 4 \times
6}}{2} = 2
\end{equation}

and so we have solved
equation~\eqref{eq:solve}
```

$$x^2 - 5x + 6 = 0 \tag{1}$$

$$x_1 = \frac{5 + \sqrt{25 - 4 \times 6}}{2} = 3 \tag{2}$$

$$x_2 = \frac{5 - \sqrt{25 - 4 \times 6}}{2} = 2 \tag{3}$$

and so we have solved equation 1

Here, notice the *eq:* prefix in the label — and that the label is placed soon after the beginning of the math mode. To reference a formula, an environment with counter would have to be used. Most of the times, you will be using the `equation` environment, as that's usually the best choice for one-line formulae whether you are using `amsmath` or not.

eqref

The `amsmath` package adds a new command for referencing formulae; it is `\eqref{}`. It works exactly like `\ref{}`, but adds parentheses so that instead of printing a plain number as *5*, it will print *(5)*. This can be useful to help the reader distinguish between formulae and other things, without the need to repeat the word "formula" before any reference.[4] Its output can be changed as desired; for more information see the `amsmath` documentation.

tag

The `\tag{`*eqnno*`}` command is used to manually set equation numbers, where *eqnno* is the text string you want to display instead of the usual equation number. It is normally better to use labels, but sometimes hard-coded equation numbers might offer a useful work-around — such as the case where you want to repeat an equation that has already been used before (e.g. `\tag{\ref{eqn:before}}`).

numberwithin

The `amsmath` package adds the `\numberwithin{countera}{counterb}` command, which replaces the simple `countera` with a more sophisticated `counterb.countera`. For example, `\numberwithin{equation}{section}` in the preamble will prepend the section number to all equation numbers.

cases

The `cases` package adds the `\numcases` and the `\subnumcases` commands, which produce multi-case equations with a separate equation number and a separate equation number plus a letter, respectively, for each case.

The `varioref` package

The `varioref` package introduces a new command called `\vref{}`. This command is used exactly like the basic `\ref`, but it has a different output according to the context. If the object to be referenced is in the same page, it works just like `\ref`; if the object is far away, it will print something like "5 on page 25", i.e. it adds the page number automatically. If the object is close, it can use more refined sentences such as "on the next page" or "on the facing page" automatically — according to the context and the document class.

This command has to be used very carefully. It outputs more than one word, so it may happen that its output falls on two different pages. In this case, the algorithm can get confused and cause a loop.

For example, you could label an object on page 23 and the `\vref` output could happen to stay between page 23 and 24. If it were on page 23, it would print like the basic `ref`, if it were on page 24, it would print "on the previous page", but it is on *both*, and this may cause some strange errors at compiling time that could be very difficult to fix.

And while for small documents, these situations might happen very rarely, for long documents spanning hundreds of references, these situations are more likely to happen. One way to avoid these problems during document preparation is to use the standard `ref` all the way through at first, convert all to `vref` when the document is close to its final version — before making the adjustment to fix any possible problem.

The `hyperref` package

autoref

The `hyperref` package introduces another useful command; `\autoref{}`. This command creates a reference with additional text corresponding to the target's type, all of which will be a hyperlink. For example, the command `\autoref{sec:intro}` would create a hyperlink to the `\label{sec:intro}` command, wherever it is. Assuming that this label is pointing to a section, the hyperlink would contain the text "section 3.4", or similar (the full list of default names can be found here (https://packages.oth-regensburg.de/ctan/macros/latex/contrib/hyperref/doc/hyperref-doc.html#x1-140004)). Note that, while there's an **\autoref*** command that produces an unlinked prefix (useful if the label is on the same page as the reference), no alternative **\Autoref** command is defined to produce capitalized versions (useful, for instance, when starting sentences); but since the capitalization of autoref names was chosen by the package author, you can customize the prefixed text by redefining `\`*type*`autorefname` to the prefix you want, as in:

```
\def\sectionautorefname{Section}
```

This renaming trick can, of course, be used for other purposes as well.

- If you would like to have a hyperlink reference without the predefined text `\autoref{}` provides, then you change this with a command such as `\hyperref[sec:intro]{Appendix~\ref*{sec:intro}}`. Note that you can disable the creation of hyperlinks in `hyperref`, and just use these commands for automatic text.

- Keep in mind that the \label *must* be placed inside an environment with a counter, such as a table or a figure. Otherwise, not only will the number refer to the current section (as mentioned above), but the name might refer to the previous environment with a counter as well. For example, if you put a label after closing a figure, the label will still say "figure n", on which n is the current section number.

nameref

The `hyperref` package also automatically includes the `nameref` package, and a similarly named command. It is similar to `\autoref{}`, but inserts text corresponding to the section name, for example.

Input:

```
\section{MyFirstSection} \label{sec:marker}
\section{MySecondSection}
In section~\nameref{sec:marker} we defined...
```

Output:

In section MyFirstSection we defined...

Anchor manual positioning

When you define a `\label` outside a figure, a table, or other floating objects, the label points to the current section. In some cases, this behavior is not what you'd like and you'd prefer the generated link to point to the line where the `\label` is defined. This can be achieved with the command `\phantomsection` as in this example:

```
%The link location will be placed on the line below.
\phantomsection
\label{the_label}
```

The `cleveref` package

The `cleveref` package introduces the new command `\cref{}` which includes the type of referenced object like `\autoref{}` does. The alternate `\labelcref{}` command works more like standard `\ref{}`. References to pages are handled by the `\cpageref{}` command.

- Use `\crefrange{}{}` and `\cpagerefrange{}` commands for start and end label in either order and to provide a natural language (`babel` enabled) range.
- Use `\cref{}` command for multiple or single references (on single line), it will sort them and group into ranges automatically in very convenient format like `` ` item 2 to 4 and 6 to 19.` ``.[5]
- Use `\labelcref{}` where the numbers would be a references to the labels and [sorted] output as in previous paragraph

The format can be specified in the preamble.

Interpackage interactions for `varioref`, `hyperref`, and `cleveref`

Because `varioref`,`hyperref`, and `cleveref` redefine the same commands, they can produce unexpected results when their `\usepackage` commands appear in the preamble in the wrong order. For example, using `hyperref`,`varioref`, then `cleveref` can cause `\vref{}` to fail as though the marker were undefined.[6] The following order generally seems to work:

1. `varioref`
2. `hyperref`
3. `cleveref`[6]

See also

- LaTeX/Glossary

Notes and References

1. Advantages of LaTeX — Automatic Numbering and Robust Citation System (https://mathvault.c a/latex-guide/#Automatic_Numbering_and_Robust_Citation_System)
2. http://www.ctan.org/tex-archive/macros/latex/contrib/hyperref/README
3. Documentation of the `caption` package — 6.5 hyperref

4. TeX StackExchange — What is the difference between \eqref and \ref? (https://tex.stackexchange.com/questions/107422/what-is-the-difference-between-eqref-and-ref)
5. TeX Blog — Cleveref, a clever way to reference in LaTeX (https://texblog.org/2013/05/06/cleveref-a-clever-way-to-reference-in-latex/)
6. Tests done under report class http://tex.stackexchange.com/questions/139459/vref-and-input-command

---

Retrieved from "https://en.wikibooks.org/w/index.php?title=LaTeX/Labels_and_Cross-referencing&oldid=4302575"

This page was last edited on 3 July 2023, at 07:31.