The `varioref` package[*]

Frank Mittelbach

1995/10/19

Abstract

This package defines the commands `\vref` and `\vpageref` for LaTeX 2_ε. `\vref` is similar to `\ref` but adds an additional page reference, like 'on the facing page' or 'on page 27' whenever the corresponding `\label` is not on the same page. The command `\vpageref` is a variation to `\pageref` with a similar functionality. Generated strings are customizable so that these commands are usable with other languages.

1 Introduction

In many cases it is helpful when refering to a figure or table to put both a `\ref` and a `\pageref` command into the document especially when there are one or more pages between the reference and the object. Therefore some people use a command like

```
\newcommand{\fullref}[1]{\ref{#1} on page~\pageref{#1}}
```

which reduces the number of key strokes, necessary to make such a complete reference. But since one never knows where the referenced object finally falls, using such a device may result in a page reference to the current page which is disturbing and therefore should be avoided.

2 The user interface

`\vref` The implementation of `\vref` below produces only a `\ref` when reference and `\label` are on the same page. It will additionally produce one of the strings 'on the facing page', 'on the preceding page', or 'on the following page', if label and reference differ by one and it will produce both `\ref` and `\pageref` when the difference is larger. The word 'facing' is used when label and reference both fall onto a double spread. However, if a special page numbering scheme is used instead of the usual arabic numbering (e.g., `\pagenumbering{roman}`) then there will be no distinction between one or many pages off.

`\vpageref` Sometimes one wants to refer only to page number and again such a reference should normally be suppressed if we are refering to the current page. For this purpose the package defines the `\vpageref` command. It will produce the same strings as `\vref` except that it doesn't start with the `\ref` and except that it will produce the string that is saved in `\reftextcurrent` if label and reference fall onto the same page. By defining `\reftextcurrent` to produce "on this page" or something similar, we can avoid that

```
... see the example \vpageref{ex:foo} which shows ...
```

comes out as "... see the example which shows ...", which could be misleading.

You can put a space in front of `\vpageref` it will be ignored if the command doesn't produce any text at all.

But in fact `\vpageref` allows even more control. If has two optional arguments. With the first one, one can specify the text that should be used if label and

---

[*]This file has version number v1.1b, last revised 1995/10/19.

reference fall on the same page. This is very helpful if both are near to each other, so that they may or may not be separated by a page break. In such a case we usually know (!) whether the reference is before or after the label so that we can say something like

```
... see the example \vpageref[above]{ex:foo} which shows ...
```

which will then come out as "... see the example above which shows ..." if we are still on the same page, but as "... see the example on the page before which shows ..." (or something similar depending on the settings of the `\reftext..before` commands) when there was a page break in the meantime. One warning however, if you use `\vpageref` with the optional argument to refer to a figure or table, keep in mind that depending on the float placement parameters the float may show up on top of the current page and therefore before the reference even if it came after it in the source file.

But maybe you prefer to say "... see the above example" if example and reference fall onto the same page, i.e., reverse the word order. In fact, in some languages the word order automatically changes in that case. To allow for this variation the second optional argument can be used. It specifies the text preceding the generated reference if object and reference do not fall onto the same page. Thus one would write

```
... see the \vpageref[above example][example]{ex:foo}
    which shows ...
```

to achieve the desired effect.

3 Customization

The package supports all options defined by the babel package to translate the fixed strings into other languages than English. (Some languages need updating, however.) It also supports languages currently not in babel; check the section on options later on. You can also modify some or all of the strings by redefining the following commands. Backward references use `\reftextbefore` if the label is on the preceding page but invisible and `\reftextfacebefore` if it is one the facing page (i.e., if the current page number is odd). Similarly `\reftextafter` is used when the label comes on the next page but one has to turn the page and `\reftextfaceafter` if it is on the following but facing page.

In fact, `\reftextface...` is used only if the user or the document class specified two-sided printing.

Finally we have `\reftextfaraway` which is used whenever label and reference differ by more than one or when they aren't numeric. This macro is a bit different because it takes one argument, the symbolic reference string so that one cane make use of `\pageref` in its replacement text.

To allow a bit random variation in the generated strings one can use the command `\reftextvario` inside the string macros. It takes two arguments and selects one or the other for printing depending on the number of already seens `\vref` or `\vpageref` commands. As an example see the definitions of `\reftextbefore` etc. on page ??.

4 A few warnings

Defining commands like the ones described above poses some interesting problems. Suppose, for example, that a generated text like 'on the next page' gets broken across pages. If this happens it is very difficult to find an acceptable solution and in fact can even result in a document that will always change from one state to another (i.e., inserting one string, finding that this is wrong, inserting another string on the next run which makes the first string correct again, inserting ...). The current implementation of `varioref` therefore issues an error message whenever the generated text is broken across page boundaries, e.g.,

> table 5 on the current ⟨*page break*⟩ page

\vrefwarning   would would result in an error, which needs to be resolved by the user by replacing the \vref command with an ordinary \ref just before the final run. This is not completely satisfactory but in such case no solution really is. During document preparation, while one is still changing the text, such error messages can be turned into warnings by placing a \vrefwarning command in the preamble.

\fullref   At the end final a warning: every use of \vref will internally generate two macro names to keep track of the string positions within the document. As a result you may run out of name space or main memory if you make havy use of this macro on a small TeX installation. For this reason the primitive command \fullref is also provided. This command can be used whenever you know for sure that label and reference can't fall onto nearby pages.