labels.tex

Macros to print address labels and bulk letters.
Version: 1.0 (24 October 1991) (doc: 10 Sep 1991)

The macros defined in labels.tex can be used to print address labels. Such labels can be printed on normal paper, if you want to cut them yourself, or on peel-off labels. (Peel-off labels are available in various sizes for most types of printers and photo-copiers.) In order to use the macros you always have to start with `\input labels`, which reads labels.tex*. We will first explain how to typeset labels, then how to print bulk letters.

There are two slightly different ways to print labels. The first one is to say

```
\beginlabels
...
\endlabels
```

where in place of '...' you type the labels separated by empty lines. The second is to simply say

```
\labelfile{...}
```

where '...' is the name of a file that contains the labels. In either case a label consists of several lines, and labels are separated by one or more empty lines. You end your file as usual, with `\bye`. Labels.tex does not specify which font to use, which in plain TEX means that by default a 10 points Roman font is used. However, for legibility it is usually better to use a sans-serif font, for instance by writing

```
\font\sf=cmss12
\sf \baselineskip=14pt
```

In order to typeset labels, you must define the size and other attributes of labels. (There are default values, but they are unlikely to suit your needs, especially if you want to print on peel-off labels.) These attributes must be set before you call `\beginlabels` or `\labelfile`; therefore, a typical input file has the form

```
\input labels
\vlbls=5 \hlbls=2
\vlblsize=5cm
\labelfile{customers.lbl}
\bye
```

where file 'customers.lbl' has the addresses that must be typeset. The assignments to `\vlbls`, `\hlbls`, and `\vlblsize` define certain attributes. Next we describe which attributes there are.

First you have to tell TEX how labels must appear on a page. `\vlbls` must denote the vertical number of labels per page, `\hlbls` the horizontal number. `\vfirst` specifies the amount of white space between the top of the page and the top of the first label; `\hfirst` is the amount of white space to the left of the first label. Dimensions `\vinter` and `\hinter` specify the white space between labels.

Then you have to specify the size of each label. `\vlblsize` and `\hlblsize` specify the vertical and horizontal sizes of a single label. `\vindent` is the white space between the top of the label and the baseline of the first line. (`\vindent` is similar to TEX's `\topskip` parameter.) `\hindent` is the amount of white space between the left edge of a label and the text of the label.

The parameters mentioned above are sufficient to typeset labels. There are however a few special parameters that are sometimes useful. By default, there is no outline around the labels. That is suitable for printing on peel-off labels, but inconvenient for use with previewers or plain paper. You can set `\lbloutline` to the thickness of the lines you want around your labels, e.g., `\lbloutline=0.5pt`. The outlines are drawn at the outside of the label (i.e., in the space between the labels).

If there is too much text on a label, an error message is given; for reference, the message contains the input text corresponding to the label. A label with an error is not typeset. When you typeset labels,

---

* To use labels.tex you also need Midnight/dolines.tex and Midnight/loop.tex, which are input automatically.

by default a file 'lblerror.tex' is created, which has the text of all the labels that did not fit. This makes it convenient to process those labels again, e.g., with `\labelfile`. However, if you set `\erroraction=0` before you start typesetting labels, file 'lblerror.tex' is not created. Setting `\erroraction=1` does not only suppress generation of the error file, it also forces printing of the labels that have errors. Any other value of `\erroraction` causes the default behavior.

The error messages refer to the natural size of a label. If `\vindent` and `\hindent`, which are glue parameters, have shrink components, it is possible that the text will actually fit on the label. However, usually these indentations have no stretch or shrink components.

There are two token registers, `\beforelbl` and `\afterlbl`, that can be used to put additional text on each label. They contain ⟨vertical material⟩ that is inserted just before and after the text of the label, respectively. By default, these sequences are empty. One can for instance specify

```
\beforelbl={\line{---Confidential---\hfil}\smallskip}
```

to put "—Confidential—" at the beginning of each address. *Note:* Since addresses are normally read from bottom to top, the post office prefers it if you don't put text after the address, unless it is part of the address. E.g., using `\afterlbl={\noindent USA}` is ok, but use `\beforelbl` if you want to add an "attention:…" line to each label.

If you want to use the labels for something else than addresses, you may want to center the text on the label. Horizontal centering can be obtained with `\hindent=0pt plus1fil`. Note that this centers the text according to the widest line; it does not center each line separately. Vertical centering is a bit more complicated, because of the way TeX stacks boxes vertically. The following centers the label text vertically:

```
\count255=\baselineskip
\vindent=\count255sp plus1fil
```

This works because the actual vertical indentation is `\vindent` minus `\baselineskip`, since TeX automatically inserts `\baselineskip` glue.

Within a label, you can type '`&`' and '`#`' for '&' and '#'. (In TeX, you usually have to type '`\&`' and '`\#`' to get these symbols.) You can use macros in a label, for instance for font changes. A line of a label must contain balanced braces; a '`%`' sign will concatenate the next line. (Usually, labels do not contain macros, nor braces, nor percentage-signs.) If for some reason you want an empty line in the middle of a label, write a 'tie' ('`~`') on that line. (The `~` is replaced by a space in the output.)

Address labels are often used to send an identical letter to each addressee. The command `\bulk` makes it easy to print the address on each letter. You need two files, one with all the addresses, and one with the standard letter. Say that they are called 'addr.tex' and 'letter.tex', respectively. The command

```
\bulk{addr}{letter}
```

will typeset a copy of the letter for each address. In the letter you can use the control sequence `\bulkaddress`. This stands for a vbox that contains the address to which the letter corresponds. (The vbox contains an hbox for each line of the address; parameters like `\vindent` and `\hlblsize` are not important when `\bulk` is used.) Since 'letter.tex' is processed many times, it is best to put any definitions that apply to all letters in the file that contains the `\bulk` command, rather than in the letter itself. Each letter is read within a group, so all non-global definitions and assignments in the letter are local to the letter. Also note that the vbox containing the address is typeset before the letter is read. The `\bulk` command does not generate address labels; for that you have to use `\labelfile{addr}`.