

babeltools

Fixing (some) babel annoyances

Javier Bezos

October 14, 2010

(Beta.)

The `babel` package introduces some changes in the \LaTeX kernel which are not strictly necessary (perhaps just convenient), but have some unwanted side-effects. Oddly enough, most of these changes cannot be disabled, with a few exceptions (eg, `activeacute` and `activegrave`, but note there are not `activecaret` or `activetilde`).

With `babeltools` you can modify the `babel` behaviour by means of a set of package options, with a few macros serving as tools for specific purposes. This package must be loaded **before** `babel`.

1 Package options

shorthands=off The shorthands mechanism is turned off. As some languages use this mechanism for tools not available otherwise, a macro `\babelshorthand` is defined, which allows using them; see below.

shorthands=... The shorthands mechanism is on, but the only shorthands activated are those given, like, eg:

```
\usepackage[shorthands=:;!]{babeltools}
```

If `'` is included, `activeacute` is passed to `babel`; if `‘` is included, `activegrave` is passed. Active characters (like `~`) should be preceded by `\string` (otherwise they will be expanded by \LaTeX before they are passed to the package and therefore they will not be recognized).

adaptive By default, all `babel` shorthands are active and live from start to end of documents. You can deactivate them by hand with `\shorthandoff`, but this is cumbersome and you cannot use it “just in case” (if the character is not a shorthands an error is raised, instead of ignoring silently the redundant setting). That means you have to accept a character like `:` is active in an English document even if you need it for just a few quotations in French. The default engine may be replaced by a new one which adapts the shorthands behaviour to the context in the following way:

1. In math mode, while still shorthands, they behave always like the corresponding normal char. Things like `$a \mathrel{x'} b$` work as expected.
2. When the language is switched, shorthands chars are made normal or active, as appropriate. Thus, `:` or `?` are active only in `french`.

nocrossrefs `newlabel`, `ref` and `pageref` are not redefined. You cannot use shorthands in labels.

nocitations `nocite`, `bibcite` and `bibitem` are not redefined. You cannot use shorthands in labels.

langcaptions Captions are redefined if possible so that an intermediate macro `\lang...name` is used, eg, `\germanchaptername`. Somewhat experimental.

noconfig Config files are not loaded, so you can make sure your document is not spoiled by an unexpected `.cfg` file.

Babel tweaks several packages for shorthands to be accepted. At the time of this writing they are `cite`, `natbib`, `varioref` and `hhline` (the latter for the colon). If you don't use shorthands, these redefinitions are unnecessary and inconvenient. If you give a list of shorthands and it doesn't include `:`, then `hhline` is left untouched. The other packages are handled by `nocrossrefs` and `nocitations`, as appropriate.

2 Macros

`\babelshorthand` Use a shorthand, even with `shorthands=off` or not listed in it, eg, `\babelshorthand{"u}` or `\babelshorthand{:}`. You can conveniently define your own macros.

3 Known limitations

If a class loads `babel` with a language, you cannot use `babeltools`. However, if a class loads `babel` without loading any language, you can use it. Languages given in `\documentclass` work as expected (provided the class doesn't load it, of course).