

The `keys3opts2` package*

Using `keys3` for $\text{\LaTeX} 2_\epsilon$ options

Joseph Wright[†]

2009/05/03

1 Package options

The methods that $\text{\LaTeX} 3$ will use to process package options are not clear at the present time. However, in order to fully test the `expl3` bundle fully, it is useful to be able to work with the $\text{\LaTeX} 2_\epsilon$ system but using $\text{\LaTeX} 3$ methods as far as possible.

The `keys3` package provides an experimental method for managing key–value input using `expl3` syntax. It is self-contained and does not use any $\text{\LaTeX} 2_\epsilon$ methods. On the other hand, `keys3opts2` is intended as a “bridge” between the $\text{\LaTeX} 2_\epsilon$ kernel and `keys3`. It allows keys to be created using `keys3` and processed by $\text{\LaTeX} 2_\epsilon$ as package or class options.

1.1 functions

`\keys_options_ltx:n` `\keys_options_ltx:n` $\langle module \rangle$

The only macro for outside use. Processes the current option list using the currently defined keys, checking if they exist for $\langle module \rangle$.

`\keys_options_ltx_cls:n`
`\keys_options_ltx_pkg:n` `\keys_options_ltx_cls:n` $\langle option \rangle$

Helper macros for processing class and package options, respectively.

`\keys_options_ltx_strip_eq:w` `\keys_options_ltx_strip_eq:w` $\langle option \rangle$ `\q_stop`

Used to strip “=” out of options.

`\l_keys_options_ltx_clist` Storage area for option list.

*This file has version number 93, last revised 2009/05/03.

[†]E-mail: joseph.wright@morningstar2.co.uk

1.2 Implementation

Almost everything here is based heavily on `pgfopts`, which is itself a port of the appropriate code from `kvoptions` to work with `pgfkeys`.

As this package is an add-on to `keys3`, the same internal module name (`\keys_`) is used here as in the main package. The lead off as usual.

```

1 <*package>
2 \ProvidesExplPackage
3   {\filename}{\filedate}{\fileversion}{\filedescription}
4 \RequirePackage{keys3}

```

`\l_keys_options_ltx_clist` A simple storage area.

```

5 \clist_new:N \l_keys_options_ltx_clist

```

`\keys_options_ltx:n` As this is a minimal testing package, a lot of the complexity of the normal L^AT_EX options is ignored. Here, the package insists a path is given for the keys. First, a path may need to be added to the argument given. The path we need to process is then stored in `\l_keys_current_key_tl`.

```

6 \cs_new:Nn \keys_options_ltx:n {
7   \tl_set:Nx \l_keys_current_key_tl {#1}
8   \keys_path_add:N \l_keys_current_key_tl
9   \clist_clear:N \l_keys_options_ltx_clist
10  \cs_if_eq:NNTF \@currentx \clsextension {
11    \cs_if_free:cF { opt@ \@currname . \@currentx } {
12      \clist_map_function:cN { opt@ \@currname . \@currentx }
13        \keys_options_ltx_cls:n
14    }
15  }{
16    \cs_if_free:NF \@classoptionslist {
17      \clist_map_function:NN \@classoptionslist \keys_options_ltx_pkg:n
18    }
19    \cs_if_free:cF { opt@ \@currname . \@currentx } {
20      \exp_args:NNc \clist_concat:NNN \l_keys_options_ltx_clist
21        { opt@ \@currname . \@currentx } \l_keys_options_ltx_clist
22    }
23  }
24  \clist_put_left:Nn \l_keys_options_ltx_clist { #1 /.cd: }
25  \exp_args:NV \keys_manage:n \l_keys_options_ltx_clist
26  \AtEndOfPackage { \cs_set_eq:NN \@unprocessedoptions \scan_stop: }
27 }

```

`\keys_options_ltx_cls:n` This macro checks if the current class option is a known key. To do that, any “=” part has to be removed first.

```

28 \cs_new:Nn \keys_options_ltx_cls:n {
29   \keys_if_cmd_exist:nTF {
30     \l_keys_current_key_tl / \keys_options_ltx_strip_eq:w #1 = \q_stop
31   }{
32     \clist_put_right:Nn \l_keys_options_ltx_clist {#1}

```

```

33   }{
34     \clist_put_right:Nn \@unusedoptionlist {#1}
35   }
36 }

```

`\keys_options_ltx_pkg:n` For packages, options need to be removed from the global list.

```

37 \cs_new:Nn \keys_options_ltx_pkg:n {
38   \keys_if_cmd_really_exist:nT {
39     \l_keys_current_key_tl / \keys_options_ltx_strip_eq:w #1 = \q_stop
40   }{
41     \clist_put_right:Nn \l_keys_options_ltx_clist {#1}
42     \clist_remove_element:Nn \@unusedoptionlist {#1}
43   }
44 }

```

`\keys_options_ltx_strip_eq:w` A simple method to remove an equals sign plus any trailing data.

```

45 \cs_new:Npn \keys_options_ltx_strip_eq:w #1 = #2 \q_stop {#1}
46 </package>

```