

luaoptions

0.9

Fr. Jacques Peron

Urs Liska

2026/05/20

Introduction

luaoptions is a LuaLaTeX package providing extensive support for handling *options*, on package level and locally. It is hosted at <https://github.com/lualatex-tools/luaoptions> and maintained by Jacques Peron (cataclop@hotmail.com) and Urs Liska (git@ursliska.de). The package was originally developed as part of the lyluatex package¹ but has now been extracted as a standalone package because we realized it can be a useful tool for arbitrary LuaLaTeX packages having to deal with *options* – both *package options* and *optional macro arguments*. A lyluatex installation should not be a dependency for packages that want to make use of option handling but are not interested in music engraving ...

luaoptions allows the declaration of sets of options, along with defaults, expected/allowed values and limited type checking. These options can be enforced as package options, changed at any point during a document, or overwritten locally by optional macro arguments. It is also possible to instantiate an Options object as an independent Lua object, without linking it to a package. luaoptions can be used to enforce and prepopulate options, or it can be used to simply handle the parsing of optional key=value arguments into proper Lua tables.

The main use of luaoptions is to define package options as follows:

```
\RequirePackage{luaoptions}
\directlua{
  local _opt = lua_options
  lua_options.register('myoptions', {
    ['option'] = {'default value', 'other value', ''},
    ['otheroption'] = {'default', validator},
    ['yetanotheroptions'] = {},
  })
}
```

validator is a function that returns true when the defined option is valid, false otherwise. luaoptions itself contains predefined such functions.

After that, defined options are available in a lua table, which can be used by the mean of the client command:

¹<https://github.com/jperon/lyluatex>

```
local myopts = lua_options.client('myoptions')
tex.sprint(myopts.option)
```

This initial release does *not* include documentation beyond this short introduction; please refer to the docstrings in `luaoptions.lua` and `luaoptions-lib.lua` for more information.

For the time being it is possible to look into how some of our own packages make use of `luaoptions`:

- <https://github.com/jperon/lyluatex/blob/master/lyluatextools.sty>
- <https://github.com/jperon/lyluatex/blob/master/lyluatex.lua>
- <https://github.com/lualatex-tools/luaformatters/blob/master/luaformatters.sty>
- <https://github.com/lualatex-tools/luaformatters/blob/master/luaformatters.lua>
- <https://github.com/lualatex-tools/luaformatters/blob/master/submodules/luaformatters-formatter.lua>