The externalize package

Lukas Heindl
oss.heindl+latex@protonmail.com

v0.0.2 from 2024/02/28

Abstract

tikzexternal is a library to build parts (pictures[1]) of a document separately. If you don't specify names for these pictures, tikzexternal will automatically generate some. This has the drawback that if you reorder or insert pictures, every picture whose name changed needs to be rebuilt. This library uses the md5 checksum of the content as name for a picture. By doing so it implicitly deduplicates the pictures.

1 Basic Usage

- Load the package: `\usepackage{externalize}`
- Configure:
  `\directlua{externalize.configure{realjobname="main",␣prefix="figures/"}}`
- Externalize tex-code: `\directlua{externalize.handle_string("tex-code")}`

After each run we check if there were any unused files (generated by this package). In case there are, they are being removed to reduce disk usage.

Be aware that `handle_string` needs an unexpanded string and that the decision wether to rebuild is also just done on that unexpanded tex-code. Thus, if your picture depends on some state outside of the picture, the rebuild decision might be wrong.

1.1 Setting names

You can pass an additional 2nd argument to `handle_string` as `name` of that picture. If you set `\jobname` to that name, only that picture will be built. This might be useful when tikzing some image but you want to avoid typesetting the whole document all the time.

The specified name is only being used if the `\jobname` matches that name. In all other cases, the classic name (with the hash) is still being used.

Because of the special usecase of this, cleanup is deactivated when running with `\jobname` set to a manually set name.

[1] Note: despite using the word "picture", you can use this with arbitrary tex-code

1.2 Externalize complete files

If you have the code for a picture already in a seperate file, you can simply use `\directlua{externalize.handle("filename")}` instead of `\input`. Just like `handle_string`, you can pass an additional 2nd argument specifying a custom name.

2 Configuration

Configuration can be done via `\directlua{externalize.configure{<config␣keys>}}`. The following keys are available:

Option	Description
`realjobname`	Needed so that we can detect if the whole document is being processed or just a single picture.
`prefix`	This will be the prefix where/how the single built pictures are being stored. You might want to use something like `figures/file-prefix` here to store all built pictures in a separate directory. Note that if you do so, this/these directory/directories have to exist beforehand.
`pre/post`	As this library requires you to pass the content unexpanded, it might prove diffucult to wrap this properly. You can use these keys to always prepend/append code to the content of the picture. Maybe something like `\begin {tikzpicture}`/`\end {tikzpicture}`

3 Issues

- tikzexternal will automatically (try to) externalize tikzpicture environments and tikz commands
- apparently using `\directlua` and the `external` table doesn't work like outlined here -> write wrapper (might get ugly since verbatim is needed) tex macros with pgfkeys! => quick fix is to simply load the package as
  `\directlua{externalize␣=␣require␣"externalize.lua"}`
  `\AtEndDocument{\directlua{externalize.clean_up()}}`

4 Still todo

- figure out how to properly build the command which builds the individual picture

5 Example

5.1 Wrap externalize with a command

```
\usepackage{externalize}
\directlua{externalize.configure{
```

```
pre = [[\unexpanded{\tikzpicture}]],
post= [[\unexpanded{\endtikzpicture}]],
}}
\NewDocumentCommand{\externTikz}{+v}{%
\directlua{externalize.handle_string([[#2]])}%
}
```

In addition you can of course configure the underlying `tikzexternal` if you like to.

6 Implementation

```
1 \RequirePackage{pgf}
```

load lua package

```
2 \directlua{externalize = require "externalize.lua"}
```

clean up unused (previously) externalized pdfs to not accumulate lots of unused pdfs

```
3 \AtEndDocument{\directlua{externalize.clean_up()}}
```

Change History