

The lua-check-hyphen package

Checking your hyphenated words

Patrick Gundlach
patrick@gundla.ch

Version 0.4
April 2, 2016

Contents

1	Introduction	1	6	Final document	3
2	Usage	1	7	Changes	3
3	Advanced usage	1	8	Bugs	4
4	Visual marker	2	9	License	4
5	Options to <code>\LuaCheckHyphen</code>	2			

1 Introduction

TeX is pretty good at hyphenating words. But a computer algorithm is not perfect, sometimes we wish to manually check all used hyphenations in a document. This package helps you on this task.

2 Usage

This package only works with LuaTeX. To activate it a

```
\usepackage{lua-check-hyphen}
```

is enough.

When you process your document with LuaTeX, a file with the suffix `.uhy` is created with that contains every hyphenated word. You can check that file for hyphenation errors. With the package option `nofile`, you can tell the package not to write the external file. In any case the list is written to the log file.

3 Advanced usage

With longer documents it becomes tedious checking for the same hyphenated words over and over. Therefore this package allows you to create a whitelist of *accepted hyphenation*. This whitelist is a simple textfile where the words are separated by

whitespace and the allowed hyphenation is marked with a dash (-). You can use the output file .uhy as an input for the whitelist file. You load the file in your document with

```
\LuaCheckHyphen{whitelist=myfile.txt}
```

and a sample file looks like this:

```
seren-ity  
morn-ings  
posses-sion
```

or

```
seren-ity morn-ings posses-sion
```

The format is not important, the entries must be separated by whitespace.

Words in this whitelist won't be reported. lua-check-hyphen assumes that these are valid break points.

You can have more than one whitelist file by giving the filenames separated by comma:

```
\LuaCheckHyphen{whitelist={file1.txt,file2.txt,file3.txt}}
```

4 Visual marker

You might want to use the PDF viewer to check for incorrectly hyphenated words. To enable visual feedback, just pass the mark option to the \usepackage or to the \LuaCheckHyphen command:

```
\usepackage[mark]{lua-check-hyphen}  
% or:  
\LuaCheckHyphen{mark}
```

This marks all non-whitelisted used hyphenation points in the document with a black square (not to be confused with overfull boxes):

```
A wonderful seren-  
ity has taken posses-  
sion of my entire soul,  
like these sweet morn-  
ings of spring which I  
enjoy with my whole  
heart.
```

5 Options to \LuaCheckHyphen

For the true/false options you don't need to write the =true part. All options except for the `whitelist` can be passed directly to the package:

```
\usepackage[mark,nofile]{lua-check-hyphen}
```

All options can be set in `\LuaCheckHyphen`.

```
\LuaCheckHyphen{mark,nofile,whitelist=myfile.txt}
```

`compact=true, false` default: true

Words with different hyphenation occurrences are listed only once.

`mark=true, false` default: false

Create visual markers when an unknown hyphenation encountered.

`nofile=true, false` default: false

Don't write to an external file (`.uhy`).

`file=true, false` default: false

Don't do anything. This is for the last document run, no marks will be generated.

`whitelist=<files>`

List of files, separated by “,”. You need to wrap the filenames in braces `{...}` so that the comma won't be seen as a option separator.

6 Final document

When you pass the Option `final` to the `\documentclass`, the `\usepackage` or the `\LuaCheckHyphen` command, all processing is disabled.

7 Changes

0.4 2016-04-01

Compact representation of hyphenated words that occur more than once with different hyphenation positions (option `compact=true`).

Sort `uhy` entries alphabetically and only use lowercase words in the file.

0.2 2012-06-22

Fix ligature counting error (github issue #1)

Add nofile option

No hyphenation list in the terminal output (log only)

License details added

8 Bugs

- Paragraphs that end with `\end{document}` are ignored. Inserting an empty line or `\par` helps.

9 License

This file (documentation and the package) is released under the terms of the MIT License. See the file `mit-license.txt` in the distribution.