

PenTeX Documentation

Neruthes

2023-05-29

Contents

1	Introduction	2
2	Installation	3
2.1	Dependencies	3
2.2	Local Install	3
2.3	Distro Admin	3
3	Basic Usage	5
3.1	Installing Header	5
3.2	Using Header from Package	5
3.3	Using Header from File	5
4	Command Line Details	6
4.1	Arguments	6
4.2	Controlling Environment Variables	6
4.3	Metadata Environment Variables	7
5	Making a Package	8
5.1	Development	8
5.2	Sharing	8
6	Standard Library	10
7	Trivial Facts	11
8	Copyright	12

Chapter 1

Introduction

PenTeX is a package management frontend for Pandoc-compatible LaTeX [header files](#).

There are occasions when writing raw LaTeX is overkill, and the Pandoc default style is not good enough, and you do not want to spend too much time on tweaking the style. PenTeX is designed to fit into the gap.

PenTeX is made for you if you are familiar with Markdown but typesetting is not what you are particularly interested in.

The source code of PenTeX is available on [GitHub](#).

Chapter 2

Installation

2.1 Dependencies

Here are the known dependencies:

- Pandoc
- TeX Live (2023 or later)

Please note that some style packages may require specific fonts. Package maintainers should declare what fonts they require.

2.2 Local Install

Clone the git repository and run the following commands:

```
./make.sh build
./make.sh install_home
```

Now you should have a portable installation in your **\$HOME**.

2.3 Distro Admin

If you are packaging this software for a distro, this section is written for you.

In your build script, run the following commands:

```
./make.sh build
./make.sh install
```

Note that you should pass `DEST` environment variable to make sure that artifact files (2 shell scripts) are properly installed to the prefixed directory. For example, you can declare `DEST=/dst114514/usr/bin` if the packager working prefix is `/dst114514`.

Chapter 3

Basic Usage

3.1 Installing Header

Suppose that you want to get a header named `pentex-example-article` which is published on NPM.

```
pentex-pkg get npm:pentex-example-article
```

3.2 Using Header from Package

```
h=npm:pentex-example-article pentex MyDoc.md
```

If package is not already available, it will be downloaded. So you do not have to install it before running this command.

Remember to run `npm update` frequently to make sure that you always use the latest version of the package.

3.3 Using Header from File

```
H=pentex-example-article.H.tex pentex MyDoc.md
```

Chapter 4

Command Line Details

4.1 Arguments

Specify the file path of the Markdown file:

```
H=example.H.tex pentex MyDoc.md
```

Pass verbatim extra arguments to Pandoc:

```
H=example.H.tex pentex MyDoc.md --toc --number-sections
```

The full list of Pandoc arguments are available in [Pandoc User Manual](#).

4.2 Controlling Environment Variables

These environment variables control the behavior of `pentex`.

- `nomake`: Set `y` to prevent automatic Pandoc invocation. Instead, print the generated command only.
- `NO_AUTO_INST`: Set `y` to prevent automatic invocation of `npm install` when working with `std`: package registry.
- `h`: Specify header from package.
- `H`: Specify header from local file path.

Extra notes:

- If both `h` and `H` are empty, the default behavior is `h=std:article-std`.

4.3 Metadata Environment Variables

These environment variables are passed as variables to Pandoc.

Variable	Default Value
<code>papersize</code>	A4
<code>fontsize</code>	11pt
<code>geometry</code>	textwidth=35em

Chapter 5

Making a Package

You know LaTeX well and you want to help other people who prefer to focus on the writing itself. Here is how your generosity turns into helpful packages.

5.1 Development

Suppose that you want to make a package with the name `mypkg`.

You should include a `mypkg.H.tex` file in the root of your package directory. PanTeX will be looking for this specific filename when the user wants to use your package.

You may publish your package on NPM after the name `mypkg`. This name must correspond to the filename of the `.H.tex` file.

5.2 Sharing

You are welcomed to share your package in the [discussion area](#) of the PanTeX repository on GitHub.

Please include its name in the `npm:mypkg` format and attach a URL to an example document PDF. One ideal source is the `README.md` file of your package.

Below is a template for you to fill. It is abstracted from the [example post](#).

```
// Title
[npm:mypkg] Lorem ipsum dolor sit amet

// Body
Name: `npm:mypkg`

Source code: https://github.com/example/mypkg

Sample PDF: https://example.com/mypkg.md.pdf
```

Chapter 6

Standard Library

PenTeX Standard Library is separately maintained as another project [pentex-std-lib](#).

To use a style in the standard library, you may use the `std` namespace as registry name.

```
h=std:article-std pentex MyDoc.md
```

Here is an incomplete list of available styles in the `std` registry:

- `article-std`
- `article-tech`

Chapter 7

Trivial Facts

- We use `xelatex` at all times. If you want to set a specific value for `--pdf-engine`, you are professional enough to run `pandoc` commands without PenTeX.

Chapter 8

Copyright

Copyright © 2023 Neruthes.

The source code of the PenTeX software suite is published with [GNU GPL 2.0](#).

This document itself is published with [CC BY-SA 4.0](#).