Project template

August 2, 2023

Abstract

1 Introduction

This is a short guide to help you setup your latex projects. It covers:

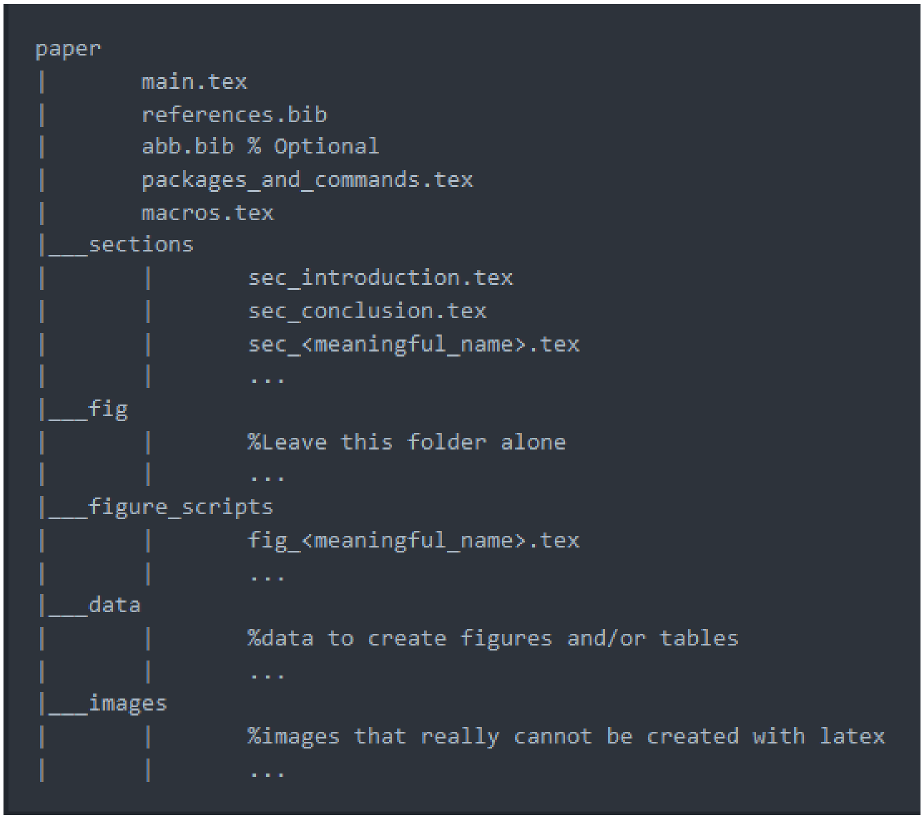
Contents

The guide also tries to provide reasons for why it is good to follow these guidelines, in form of Why-notes. If you're not interested in the reasons just skip over them.

2 Project structure

Project structure Figure 1 shows an overview of the project structure.

Why: If everyone in the group follows the same structure it is easier to collaborate and you don't need to learn a new system all the time.



```
paper
|       main.tex
|       references.bib
|       abb.bib % Optional
|       packages_and_commands.tex
|       macros.tex
|___sections
|       |       sec_introduction.tex
|       |       sec_conclusion.tex
|       |       sec_<meaningful_name>.tex
|       |       ...
|___fig
|       |       %Leave this folder alone
|       |       ...
|___figure_scripts
|       |       fig_<meaningful_name>.tex
|       |       ...
|___data
|       |       %data to create figures and/or tables
|       |       ...
|___images
|       |       %images that really cannot be created with latex
|       |       ...
```

Figure 1: The structure your project should follow.

main.tex: This is the main file of your document. Keep it as clean as possible. Conferences generally provide a template use this template to create the main file. Ideally only add the following in front of `\begin{document}`:

```
%Set to 1 to compile Figures
\def\compileFigures{0}
\newcommand{\filename}{main}
\newcounter{figureNumber}

\input{packages_and_commands}
\input{macros}
```

Don't write the sections in this file just use `\input{sections/sec_<section_name>}`.

Why: There will be multiple versions of your paper (submissions to different workshops, conferences, Arxiv, university events, etc.). Each requires a different template and you have to merge this template with your main.tex file. The simpler the main.tex file the better.

package_and_commands.tex: In this file you put all the `\usepackage{}` imports and code for more complicated commands, which you might use in multiple projects.

Why: If you found a useful package or spent time creating a complicated command you want to reuse it.

macros.tex: This file contains all your project-specific macros. Use macros as much as possible. For example, instead of writing `\mathbb{R}` every time for the real numbers create a macro `\newcommand{\R}{\mathbb{R}}`. Create macros when referring to existing algorithms, data sets, and names of things. When you go through your files with Ctrl + f trying to change every instance of a declaration you decided to change the notation for, you should have created a macro for that.

Why: Macros are often project specific having all of them in a small file makes it easier for collaborators to find them.

references.bib: The file containing all your citation references. See in the Citations section on how to create the citations.

Why: Keep the main.tex file clean.

sections: Each section of your paper should be in a separate .tex-file in this folder, give them meaningful names that start with `sec_`.

Why: This simplifies the main file and makes it easier for people to work on different sections simultaneously.

fig: The figures generated by your scripts will be saved in this folder. Add everything but the .pdf-files to .gitignore.

Why: Compiling figures can get time-consuming and sometimes requires specific settings. People not working on the figures can just input them as .pdf See Figures for more information.

figure_scripts: Each figure you create with TikZ or PGFPlots should be in a separate file with a meaningful name in this folder (starting with `fig_`).

Why: The code to generate figures can get long and make it harder to work on the text. Outsourcing the scripts makes the files cleaner.

data: The data used in your graphics.

Why: When you rerun some experiment you only need to replace the data.

images: This folder contains images (e.g. .png, .jpg.) that are used in the paper. It should not contain graphics you generated somewhere else, but really only pictures.

Why: Keep the main folder clean.

3 Naming conventions and references

Give everything meaningful names and create labels `\label{<name>}` for sections, figures, and tables as soon as you create them.

Use `sec:<meaningfulname>` , `fig:<meaningfulname>` , `tab:<meaningfulname>`, `eq:<meaningfulname>` when creating labels for sections, figures, tables and equations respectively.

A meaningful name is a name that tells the reader (i.e. future you or a collaborator) what the label refers to. For sections the section title is a good start. Avoid using numbers and cryptic letter combinations.

Why: You want others and future you to be able to understand what the label means.

4 Citations

5 Figures

6 Tables

7 Generating diff files

8 Useful Tools