A LATEX PACKAGE DEMO OR: WHAT I'VE LEARNED SO FAR

NILES JOHNSON

ABSTRACT. This is a latex demo for moderate-to-advanced users of latex. It illustrates some useful packages and habits I've developed over the years. To really use it effectively, you probably have to eventually look at the separate documentation for each individual package. Some of this is just a matter of taste (exceptionally good taste!) and some of this might seem like overkill now but become useful later.

CONTENTS

1. GENERAL COMMENTS

This document uses 4 style files that I've developed for my texing. I'll give an overview of the highlights in this document, but it's a good idea to read through the included files and see what parts you do and don't want to use.

The files are

- Environments: thm, prop, etc
- PageSetup: stuff about page layout, references
- Definitions: mostly macros

Date: 13th Apr, 2017.

2010 *Mathematics Subject Classification.* 68U15.

- DraftSetup: stuff that's only for draft documents

First, here are some basic packages that are good to include

i. babel: language setting
ii. fouriernc: font replacement including all math fonts, script and blackboard bold fonts. Such font packages are exceedingly rare.
iii. hyperref: linked references and better pdf documents
iv. geometry: beter margin management
v. watermarkdraft: does what it says
vi. showkeys: prints label keys in the margin

sec:latexmk

1.1. **latexmk.** This is a little script included with standard tex distributions. It checks file modification times and automatically runs latex as many times as necessary, including bibtex if necessary. It has a continuous mode which watches files you're currently working on and reruns on every update.

1.2. **text editor.** If you don't already have a good text editor, you should really start using one. You don't have to use all the features right away, but over time you'll appreciate it more and more. "Good" here means it has the following, at minimum:

- tex or latex mode with syntax highlighting
- support for automatically writing begin/end blocks
- support for searching .bib file and inserting citation keys
- support for finding and inserting appropriate label keys
- snippet support, for advanced macros
- indentation and text re-flowing support. 60 charachters is a generally accepted guideline for good line length.

For these, I use emacs with RefTeX, AUCTeX, and yasnippet. There are many alternatives to match your taste.

1.3. **version control.** This is beyond the scope of this document, but seriously look into it. It's *such* a good idea. I use git.

1.4. Γ **and other tex in section headings.** If at all possible, avoid tex in section headings, because these are put into the pdf table of contents, which can't process tex. But if you absolutely have to do it, use `\texorpdfstring` and an appropriate unicode alternative if you can find one. Note that you have to prerender unicode – see `PageSetup.sty`

2. CLEVEREF

sec:cleveref

The cleveref package includes the environment name with the reference, such as Theorem 2.5. This is very useful for when you need to change the environment type of a result. It is clever about referencing multiple results at once, using "and" or a range where appropriate. For example, the environments below are Definition 2.4, Displays (2.1) and (2.2), Theorems 2.5 to 2.7, Proposition 2.8, and Lemma 2.9. There are all kinds of ways to customize the list formatting; the package documentation is the best reference.

I generally prefix my labels with the environment type, to help keep them straight in my mind, but this is unnecessary.

Here's an equation

{disp1}

$$e = mc^2 \tag{2.1}$$

This is the best environment to use for displayed diagrams too, so I have cleveref just call the environment "Display".

For multiline equations, use "`align`" or "`align*`":

{disp2}

$$(x+y)^2 = (x+y)(x+y) \tag{2.2}$$
$$= x(x+y) + y(x+y)$$
$$= x^2 + xy + yx + y^2 \tag{2.3}$$

defn **Definition 2.4.** This is how to define a definition.

And for a theorem and its proof you would type:

thm:atheorem **Theorem 2.5.** *This is the statement of a theorem.*

Proof. And this shows that the statement is correct. □

lem:btheorem **Theorem 2.6.** *another theorem. Try changing it to a lemma.*

thm:ctheorem **Theorem 2.7.** *a third theorem.*

prop:foo **Proposition 2.8.** *a proposition*

lem:foo **Lemma 2.9.** *a lemma*

3. About text in TEX

3.1. **Symbols.** If you need the tex command for a symbol, the best way to get it nowadays is Detexify.

3.2. **fonts.** Develop a consistent system for how you use fonts. I like to use Zapf Chancery for named categories, such as $\mathcal{C}at$, $\mathcal{M}on\mathcal{C}at$, $\mathcal{A}b$, etc. Then I use Euler Cal for categories, such as $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, etc. I use Ralph Smith Formal Script for fancy script, such as $\mathscr{A}$, $\mathscr{B}$, $\mathscr{C}$.

3.3. **Further comments on text.** If you want to *emphasize* something in your text, use `\emph{}`. For **boldface**, *italics*, `monospace`, and SMALLCAPS, use `\textbf{}` `\textit{}`, `\texttt{}`, and `\textsc{}`.

These are modern improvements over the older `\bf` etc.

3.4. **Punctuation.** Punctuation should go outside of math mode, otherwise the spacing will be off. As in a, b, and c.

Tex puts more space after sentence-ending punctuation, so if you use a period mid-sentence, use the tilde to give a normal (and non-breaking) space. As in, e.g. this sentence.

In math-mode, a hyphen is interpreted as a minus sign. So if you want to write G-$\mathcal{C}at$ for the category of G categories, you need to use `\mbox{-}` or the macro `\mh`.

4. Diagrams

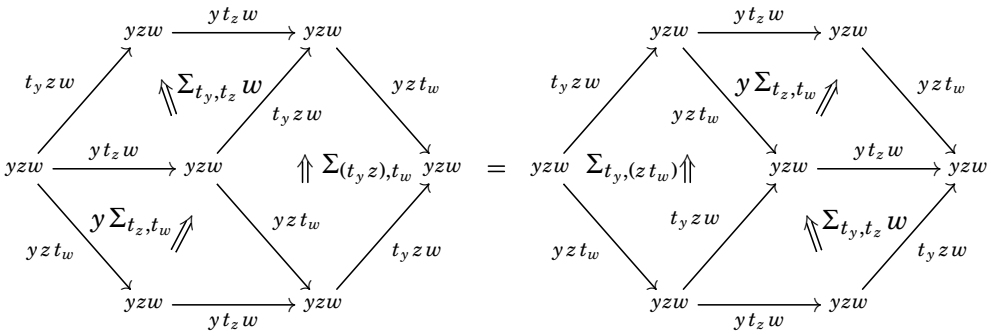
For commutative diagrams, there are two basic choices: xypic and tikz. There is plenty to say about the relative advantages and disadvantages . . . maybe another time. Whichever you use, the most common way to draw commutative diagrams is through some sort of "matrix" format: vertices are arranged on a rectangular grid, and arrows are drawn between grid coordinates. In xypic, this is the *xymatrix* environment, and for tikz this is the *tikz-cd* package. The majority of users can stop reading this section now and go search for information about these two systems. (Really. The matrix environments are good, and there are lots of good examples online. Come back here later when you decide you need more.)

4.1. **If you need more from your matrix diagram-drawing environment.** In my experience, the matrix-style drawing environments have limitations when it comes to more complex diagrams, where you want more individual control of vertex positioning and arrow type. Here are some examples in both xypic and tikz. In both formats we have the following.

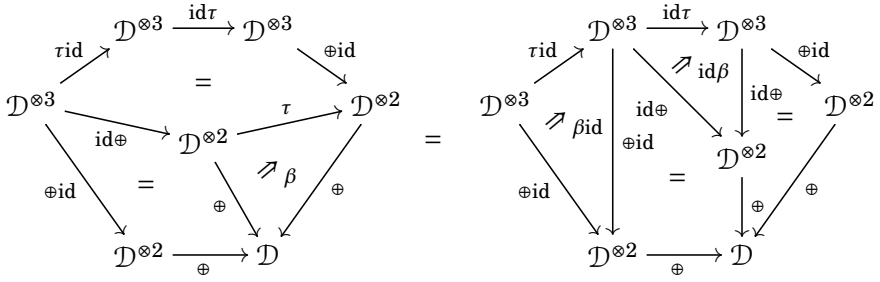
- Objects are positioned by coordinates and given labels.
- Arrows connect objects referenced by their labels; thus when you adjust objects, the arrows automatically adjust.
- Scale is controlled by independent parameters, so the entire diagram can be adjusted as necessary.

These examples can all be made without too much trouble in the matrix environments too; that's not the point. It's when *most* of your diagrams look like this that you might think about leaving the matrix environments behind.

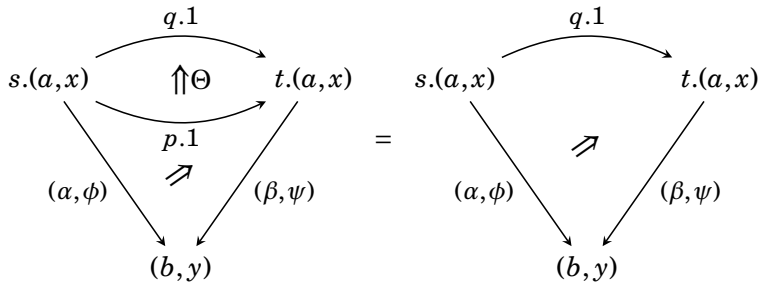
4.2. **xypic examples.** In this example, the emphasis is on 1 and 2-cells, so I made the objects smaller and the morphisms larger – that's neither necessary nor appropriate in general. Take a look at the code for the rest of the details.



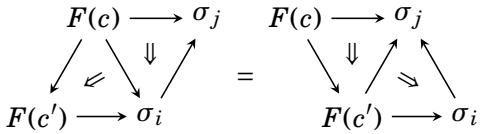
Here is another example diagram.



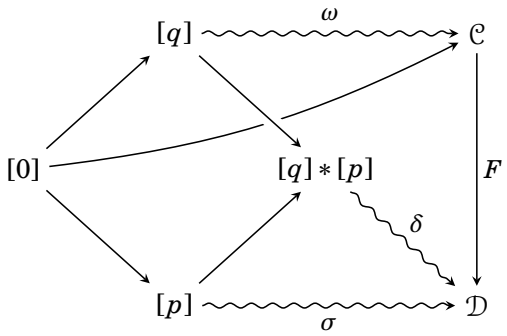
4.3. **tikz examples.** As a general drawing program, tikz is somewhat more full-featured than xypic. This means that to easily draw commutative diagrams, one needs to restrict the general functionality of tikz (for example, tikz does not draw arrowheads on connecting lines by default, but we want this for commutative diagrams). Some of these customizations are available in the tikz-cd package, but I've found that more is generally required (unless one wants to use the matrix format of the *tikzcd* environment). My adjustments are here in the tex source for this document.



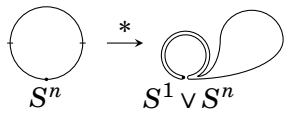
Here's another example which takes advantage of using polar coordinates instead of cartesian coordinates to position vertices.



Here's another example showing different arrow styles.



And finally, here's another tikz example I like, demonstrating how it can be used for drawings other than commutative diagrams.



5. TODONOTES

This is a package for adding marginal notes in your document. For drafts. Let's face it: for most of the time you look at a tex document, it's a draft.

such as this

you can also include inline notes

And you can make marginal comments with no line. See the package documentation for more info (such as how to change the color, etc).

such as this

6. BIBLIOGRAPHY

Here are some references to look at: [ATC, Sage, JN10, GO12, Eve91, Eve61, EKMM97, Ada74]

When you need new additions to your bibtex database, the best way to get them is from https://zbmath.org/ or http://www.ams.org/mathscinet/. Both give detailed citation info in bibtex format for easy copy/pasting. And both require a subscription (i.e., a campus connection or proxy). Zentralblatt has a more forgiving search feature,

draft mode prints a black box to mark lines that latex could not break satisfactorily. You have to reword them or do something else to handle them. Usually best to ignore until the very final stages of editing.

and lets you see the first few matches without subscription. Often enough to get the database entries you need.

I use JabRef to manage my database. Although its interface is a little clunky, I like it for 4 reasons:

i. it automatically generates cite keys according to any pattern you like. I use this pseudo-amsalpha pattern: `[authorsAlpha][year][veryshorttitle]`
ii. you can add references by copy/pasting the bibtex source
iii. it's cross-platform
iv. it automatically alphabetizes new entries to your `.bib` file, which I like for those times I edit the file directly.

This document uses a customization of the `amsalpha` bibliography style. It's additional features are:

i. First names of authors are always abbreviated.
ii. The 'key' bibtex field is respected if present, and supercedes the author field, as in [ATC, Sage]
iii. The MR number is not printed, to avoid unfair promotion over Zentralblatt
iv. The doi is printed if present, as in [JN10]
v. A new arxiv field, as in [GO12, JN10]

7. Wait, there's more!

Yes, really. Typesetting is not at all a trivial art. Just learn as you go, and keep doing that. There are other more extensive guides for latex best-practices – this document just represents what I've been able to glean from them so far.

Acknowledgments

This document began as a template for REU students at the University of Chicago. Further improvements were made by Anna Marie Bohmann. Niles Johnson made some other additions and modified it for a more advanced audience.

References

Ada1974Stable [Ada74] J. F. Adams, *Stable Homotopy and Generalised Homology*, Chicago Lectures in Mathematics, University of Chicago Press, 1974. (cit. on p. 5).

ATC [ATC] *AlgTop-Conf*, An open list for conference announcements. (cit on pp. 5, 6).

EKMM1997Rings [EKMM97] A. D. Elmendorf, I. Kriz, M. A. Mandell, and J. P. May, *Rings, modules, and algebras in stable homotopy theory*, Mathematical Surveys and Monographs, vol. 47, American Mathematical Society, Providence, RI, 1997, With an appendix by M. Cole. (cit. on p. 5).

Eve1991Cohomology [Eve91] L. Evens, *The cohomology of groups*, Oxford Mathematical Monographs, The Clarendon Press Oxford University Press, New York, 1991, Oxford Science Publications. (cit. on p. 5).

Eve1961Cohomology [Eve61] L. Evens, *The cohomology ring of a finite group*, Trans. Amer. Math. Soc. **101** (1961), 224–239. (cit. on p. 5).

GO2012Infinite [GO12] N. Gurski and A. Osorno, *Infinite loop spaces, and coherence for symmetric monoidal bicategories*, 2012. arXiv:1210.1174 (cit on pp. 5, 6).

JN2010Complex [JN10] N. Johnson and J. Noel, *For complex orientations preserving power operations, p-typicality is atypical*, Topology and its applications **157** (2010), no. 14, pp. 2271–2288. doi:10.1016/j.topol.2010.06.007 arXiv:0910.3187 (cit on pp. 5, 6).

sage [Sage] The Sage Development Team, *Sage mathematics software*, http://www.sagemath.org. (cit on pp. 5, 6).

Department of Mathematics, The Ohio State University Newark
E-mail address: niles@math.osu.edu
URL: http://nilesjohnson.net