

L^AT_EX Workshop 4: Lab Reports

Dan Parker and David Schwein

12 October 2014

Writing a lab report is by and large the same as writing any other document. The main difference is that lab reports require a few additional elements: figures with captions, cross-references, and a bibliography, among others. Today's workshop will explain how to make these elements with L^AT_EX.

1 Cross References

The reason that we number pages, figures, and other parts of a document is so that we can easily refer to them later. L^AT_EX has a simple mechanism for making cross references, which we saw last time when we discussed referencing equations.

```
\section{Introduction}
\label{sec:introduction}
Welcome to my lab report.
```

1 Introduction

Welcome to my lab report.

```
\section{Results}
See Section
\ref{sec:introduction}
for a friendly welcome.
```

2 Results

See Section 1 for a friendly welcome.

To refer to an object, you have to label it. The command for labeling is `\label{...}`, which takes the label as its argument and goes after the object you want to label. To refer back to a labeled object, use the command `\ref{...}`, which takes the label as its argument and prints the labeled object's number. Last time we used the `\eqref{...}` command, from the `amsmath` package, which acts exactly like `\ref` except that it adds parentheses around the label number.

Here are a few of the objects that L^AT_EX can label:

1. Sections and subsections
2. Equations
3. Figures and tables

We haven't yet seen how to make tables and figures in a way that allows referencing, although we will presently.

To refer to a page number, use the command `\pageref{...}`, which prints the page number that its argument appears on.

2 Graphics

Including graphics in \LaTeX is easy. Positioning them correctly is tricky because graphics tend to take up lots of space. But before we talk about how to do either of these things, let's take a step back and talk about good graphics and bad graphics.

2.1 File Formats

Broadly speaking, computer graphics fall into two categories: raster and vector. *Raster graphics* tell the computer how to draw an image by listing the color of each pixel. *Vector graphics*, in contrast, tell the computer how to draw an image by describing the image at a high level. For instance, a raster graphic would tell the computer to draw a red circle by giving it the positions of all the red pixels in the circle, while a vector graphic would tell the computer to "make a red disk with radius 1 cm in the center of the image".

The difference between the two file formats is that vector graphics are resolution-independent: you can zoom magnify a vector graphic without any loss of resolution. Raster images do not scale well and look pixelated after enough magnification. For this reason, you should use vector graphics whenever possible.

Raster File Formats	Vector File Formats
<code>.jpg</code>	<code>.eps</code>
<code>.gif</code>	<code>.svg</code> ¹
<code>.png</code>	<code>.ps</code>
<code>.tiff</code>	<code>.pdf</code>

You may need to make an exception to this rule if your image is a photograph, since photographs are by necessity raster graphics.

2.2 Figures

Including an image in a \LaTeX document requires the `graphicx` package. After loading `graphicx`, move the image into the same directory as your `.tex` file.

¹Unfortunately, \LaTeX cannot load `.svg` graphics. To include an `.svg` graphic, you must convert it into one of the other vector graphics formats.

```
\usepackage{graphicx}
. . .
\includegraphics{vector.eps}
```



The `\includegraphics` command takes one argument, the path to the image file relative to your `.tex` file. This can be used to place images into a \TeX file from other directories. For instance, if you a directory called `images` in the same directory as your `.tex` file, then you can include an image inside the `images` folder with the command `\includegraphics{images/vector.eps}`.

To make a captioned figure, use the `figure` environment.

```
\usepackage{graphicx}
. . .
\begin{figure}[h]
\center
\includegraphics{vector.eps}
\caption{An example image}
\label{fig:example_image}
\end{figure}
See Figure
\ref{fig:example_image}.
```



Figure 1: An example image

See Figure 1.

As this example shows, it is possible to label and reference figures. Make sure to place `\label` *after* `\caption`, or else the numbering will not come out right.

The `figure` environment takes an optional command which controls where the figure is placed on the page. There are three possibilities: `[h]` for *here* on the page, `[b]` for the *bottom* of the page, or `[t]` for the *top* of the page.

Positioning images can be quite tricky because \LaTeX automatically positions them for you, and it can be hard to predict or control where they will appear. If the `h`, `t`, or `b` option doesn't put your image where you want it to be, then try changing the size of the image, or the position of the `figure` environment in your code. Using references, you can always direct the reader to a figure on another page.

Finally, because changing the text of your document can change the positions of figures, remember to

Position figures only after the text of your document is complete.

You may want to try the `wrapfig` package, which places figures so that the text wraps around them. This functionality is often useful for small images that don't need a blank horizontal band, or figures that don't need much emphasis.

3 Floats

The figure environment is an example of a *floating environment*. Floating environments are different from other environments in that their position in the .tex file does not always correspond to the position of the result in the typeset document. The reason for this behavior is that L^AT_EX wants flexibility in positioning large elements of the document.

The other commonly used float environment is the `table` environment. This is *not* the same as the `tabular` environment. The `tabular` environment, which we encountered last week, is for making tables. The `table` environment is for positioning them. It uses the same basic syntax as the `figure` environment.

```
\begin{table}
\caption{Animals}
\begin{tabular}[t]{ll}
\toprule
Animal & Color\\
\midrule
Elephant & Grey \\
Lion & Yellow \\
Blue Whale & Blue \\
\bottomrule
\end{tabular}
\end{table}
```

Table 1: Animals

Animal	Color
Elephant	Grey
Lion	Yellow
Blue Whale	Blue

4 BibTeX

BibTeX is a slight extension to L^AT_EX for managing references. It's made to be easy to use and highly convenient. Citations are references are generally bothersome to do by hand, for two reasons:

1. Everyone wants a very specific, but slightly different format for citations (e.g. MLA, Chicago, etc).
2. Citations are often numbered by order of appearance, so any time you insert a new citation, all the numbers change.

BibTeX has two elegant solutions to these problems:

1. Citation formatting is automatic.
2. Citation numbering is automatic.

This is one of the best examples of the L^AT_EX philosophy; BibTeX does all the difficult formatting for you, so you can focus on content. There are three steps to using BibTeX: creating a bibliography file, citing with the `\cite` command, and typesetting for BibTeX. Let's look at them in order.

4.1 Creating a Bibliography File

BibTeX requires an external bibliography. There are several external tools to create these quickly, but it's useful to understand how they work before taking shortcuts. This takes the form of a file in the same folder as your `.tex` file that has the extension `.bib`. It's customary to call it `references.bib` or `citations.bib`, or something along those lines. The `.bib` files contains the information about each thing you want to cite.

BibTeX supports many types of bibliographic entries, including `book`, (journal) `article`, (conference) `proceedings` and many more. Let's look at a few examples:

```
@book{rudin1964principles,
  title={Principles of mathematical analysis},
  author={Walter Rudin},
  volume={3},
  year={1964},
  publisher={McGraw-Hill New York}
}

@article{bardeen1957theory,
  title={Theory of superconductivity},
  author={Bardeen, John and Cooper, Leon N and Schrieffer, J Robert},
  journal={Physical Review},
  volume={108},
  number={5},
  pages={1175},
  year={1957},
  publisher={APS}
}
```

Each entry starts of with `@` and then the name of that type of entry. This is followed by the *name* of that entry, a unique identifier that you can use to cite that entry. For the second entry above, the name is `bardeen1957theory`. After that, there several different fields that give the information needed to construct the reference. For `book`, the required fields are `title`, `author`, `publisher`, `year`. For `article`, it's also necessary to include `journal`, and highly recommended to provide `year`, `volume`, `page` so that a reader could easily find the article in question. A list of entry types and what information is required for each one is available at http://en.wikibooks.org/wiki/LaTeX/Bibliography_Management. Notice that there is a *comma* after each field. This is absolutely necessary to include.

Of course, no one wants to bother typing all this out for each citation, so let's look at easier ways to do this. Luckily, there are many good ones!

Google Scholar If you find your reference on Google Scholar, then you get the BibTeX entry for free! Under the name of the entry in Google Scholar,

there are several links: “Cited by...”, “Related Articles”, “Cite” and several more. If you click “cite”, you will get a citation for that work in several formats, one of which is BibTeX. These days, most journals have all their articles online and have a similar “generate citation” button for each article.

JabRef A (free, open source) program for managing BibTeX files. Works on Windows, Macs, and Linux. Gives a nice graphical user interface (GUI) for adding entries. It also provides an interface to search online journal databases and turn the search results into bibliography entries.

BibDesk A similar tool, that is slightly more polished and user friendly, but only works on Macs. This is included with TeXShop.

4.2 Citing in the Text

Citing from the text is very easy. Let’s look at an example.

```
A standard textbook on Analysis is \cite{rudin1964principles}.
```

```
The theory of low-temperature  
superconductivity was invented in 1957 \cite{bardeen1957theory}.
```

```
\bibliographystyle{plain}  
\bibliography{references.bib}
```

A standard textbook on Analysis is [2].

The theory of low-temperature superconductivity was invented in 1957 [1].

References

- [1] John Bardeen, Leon N Cooper, and J Robert Schrieffer. Theory of superconductivity. *Physical Review*, 108(5):1175, 1957.
- [2] Walter Rudin. *Principles of mathematical analysis*, volume 3. McGraw-Hill New York, 1964.

To cite a bibliographic entry, use the `\cite` command with the name of that entry. The command `\bibliographystyle{plain}` says how the citations should be formatted. The `plain` style will sort references by author and use squares brackets with numbers in the text. There are many other styles available, such as `unsrt`, which gives the citations in order of appearance. The `alpha` style will give citations like [Rud64]. The `natbib` package gives many more options, including `apa`. To use MLA-style citations, try the `biblatex-mla` package, which is a little more involved. Many many other formats are available, and can be found with a little googling around.

4.3 Typesetting with BibTeX

Since BibTeX has an external file, you need to typeset a little differently. After any changes to the BibTeX file, you must go through the following sequence:

1. Typset L^AT_EX normally.
2. Typeset BibTeX. This is normally a separate button or menu option.
3. Typset L^AT_EX normally.
4. Typset L^AT_EX normally *again*.

4.4 URLs and DOIs In Citations

Almost all journals are online now, so it's usually more useful to give the DOI or a link to an article than the journal issues. To do this, load the `hyperref` and `doi` packages, and add the `url` and `doi` packages to your bibtex entries. As an example:

```
@article{bardeen1957theory,  
  title={Theory of superconductivity},  
  author={Bardeen, John and Cooper, Leon N and Schrieffer, J Robert},  
  journal={Physical Review},  
  volume={108},  
  number={5},  
  pages={1175},  
  year={1957},  
  publisher={APS}  
  doi = {10.1103/PhysRev.108.1175},  
  url = {http://link.aps.org/doi/10.1103/PhysRev.108.1175}  
}
```

In some — but not all — bibliography styles, these will show up at the end of the citation. Try the `plainnat` style to get this to work (requires the `natbib` package).