# GETTING UP AND RUNNING WITH $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX

PHILIP S. HIRSCHHORN

Abstract. This is an attempt to tell you enough about LaTeX and $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX so that you can get started with it *without* having to read the book.

Contents



---

*Date*: November 16, 2008, Version 2.01.

## 1. Introduction

This is an attempt to get you up and running with $\mathcal{AMS}$-LaTeX with the least possible pain. These instructions won't be a substitute for the User's Guide, but they may get you started quickly enough so that you'll only need to refer to the guide occasionally, which should eliminate most of the pain.

The current version of $\mathcal{AMS}$-LaTeX (version 2.0) is really just an optional package for the new standard LaTeX. $\mathcal{AMS}$-LaTeX provides the document classes `amsart` and `amsbook` (see section 2.1) to replace the standard document classes `article` and `book`, and an optional package `amsmath` that can be used with the standard LaTeX document classes. Thus, using $\mathcal{AMS}$-LaTeX is really using a variety of LaTeX. If you're new to LaTeX, and these last few sentences made no sense to you at all, don't worry about it. You don't have to know what the standard LaTeX document classes are in order to use the $\mathcal{AMS}$-LaTeX replacements for them.

I'll be assuming that you have at least some experience with either plain TeX, $\mathcal{AMS}$-TeX or LaTeX, and I'll try to tell you what you need to know so that you can get started with $\mathcal{AMS}$-LaTeX *without* actually reading the LaTeX user's guide [6], or even taking much of a look at the $\mathcal{AMS}$-LaTeX user's guide [1] or the short math guide for LaTeX [5].

If you've never used *any* version of TeX, then I recommend "The not so short introduction to LaTeX $2_\varepsilon$", by Tobias Oetiker, Hubert Partl, Irene Hyna, and Elisabeth Schlegl [7]. This is intended for those with no knowledge of TeX or LaTeX, and concisely gives a description of what a LaTeX document looks like and how you type text and simple mathematics in a LaTeX document.

I've also given you a template file `template.tex`, which is an attempt to give you enough to fake your way through an $\mathcal{AMS}$-LaTeX file *almost* without even reading these instructions. I've included the text of that file in these instructions as section 12, so you might want to take a look at that now, and then just use the table of contents of these instructions to get more information on whatever in that file confuses you.

In case you haven't guessed, these instructions were printed using $\mathcal{AMS}$-LaTeX, so you can get some idea what it all looks like.

## 2. Basic LaTeX stuff

In this section, we'll describe the three commands that have to be part of any LaTeX document: `\documentclass`, `\begin{document}`, and `\end{document}`. The complete explanation of these can be found in the LaTeX User's Guide [6] or in *The not so short introduction to LaTeX 2ε* [7]. We'll also explain how to begin a new section or subsection of the paper, and how LaTeX manages to get the cross-references right (which is also the explanation of why you need to run a file through LaTeX *twice* to be sure that all the cross-references are correct).

2.1. **The `\documentclass` command.** Before you type anything that actually appears in the paper, you must include a `\documentclass` command. It's easiest to just put the `\documentclass` command at the very beginning of the file, possibly with a few lines of comments before it.

It's actually the choice of document class that determines whether you're using $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX or just plain old LaTeX. There are two document classes that are a part of $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX: `amsart` and `amsbook`. There is also the `amsmath` package that can be used with the standard LaTeX document classes. I'll only be discussing the `amsart` document class here. For the others, see the $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX User's Guide [1].

The simplest version of the `\documentclass` command is

```
\documentclass{amsart}
```

This will give you the default type size, which is 10 point type. If you'd like to use 12 point type, then you should include the optional argument `[12pt]`; this makes the command

```
\documentclass[12pt]{amsart}
```

There are at least two optional packages that might be of interest. The first is for when you want to use some of the special symbols contained in the $\mathcal{A}\mathcal{M}\mathcal{S}$-Fonts package. These are listed in `symbols.pdf`, available at

`http://www.ctan.org/tex-archive/info/symbols/math/symbols.pdf`

If you want the standard names for these symbols to be defined for your use, then you need to use the optional package `amssymb`. Thus, to use the default 10 point type and have the special symbols defined, use the commands

```
\documentclass{amsart}
\usepackage{amssymb}
```

Another widely used optional package is XY-pic, which enables you to draw commutative diagrams as part of your LaTeX file, rather than creating them with a graphics package and importing the graphics. (For commutative diagrams, see section 8). To use XY-pic, you should include the commands

```
\usepackage[all,cmtip]{xy}
\let\objectstyle=\displaystyle
```

That loads the XY-pic package and sets it so that the nodes in the diagram are, by default, in `\displaystyle`. If you'd like the default style for the nodes to be `\textstyle`, you should omit the second of those two lines.

This document uses both of those packages, and so we used the commands

```
\documentclass{amsart}
\usepackage{amssymb}
\usepackage[all,cmtip]{xy}
\let\objectstyle=\displaystyle
```

2.2. **`\begin{document}` and `\end{document}`.** Everything that is to appear in the document must appear in between the `\begin{document}` and `\end{document}` commands. There are no optional arguments for these commands, so they always look the same. Anything following the `\end{document}` command is ignored. You *are* allowed to have macro definitions (i.e., newcommands; see section 9) before the `\begin{document}`, and that's actually a good place for them, but that's about all.

2.3. **Sections and subsections.** To begin a new section, you give the command

```
\section{Section name}
```

To begin the present section, I gave the command

```
\section{Basic \LaTeX{} stuff}
```

A section number is supplied automatically. If you want to be able to make reference to that section, then you need to *label* it. Since I wanted to be able to demonstrate the cross-reference commands, I actually began this section with the lines

```
\section{Basic \LaTeX{} stuff}
\label{sec:basicstuff}
```

This allows me to say "`section~\ref{sec:basicstuff}`" and have it printed as section 2.

To begin a new subsection, you give the command

```
\subsection{Subsection name}
```

To begin the present subsection, I gave the command

```
\subsection{Sections and subsections}
```

A subsection number is supplied automatically. If you want to be able to make reference to that subsection, then you need to *label* it. This subsection was begun with the lines

```
\subsection{Sections and subsections}
\label{sec:sections}
```

so if we say "`section~\ref{sec:sections}`," it is printed as section 2.3.

Labels always take the number of the smallest enclosing structure. Thus, a `\label` command that's inside a section but *not* inside a subsection or Theorem or anything else will take the value of the section counter, while a `\label` command that's inside the statement of a Theorem will take the value of that Theorem number. For more information on this, see section 5.

2.3.1. *Yes, there are subsubsections too.* I began this subsubsection with the command

```
\subsubsection{Yes, there are subsubsections too}
```

I refuse to even experiment to see if there are subsubsubsections.

*Sections without numbers.* I began this subsubsection with the command

```
\subsubsection*{Sections without numbers}
```

and got a subsubsection that wasn't numbered. If you give the command

```
\section*{A Section Title}
```

then you'll begin a new section that will not have a number.

2.4. **Italics *for emphasis*.** If you want to use italics to emphasize a word or two, the LaTeX convention is not to switch explicitly to italics, but rather to use the command `\emph` (which means *emphasize*). This command works just like a font change command, except that it switches you *into* italics if the current font is upright and switches you *out of* italics if the current font is italics.

For example, if you type

```
The whole is \emph{more} than the sum of its parts.
```

you'll get

The whole is *more* than the sum of its parts.

but if you type

```
\begin{thm}
  The whole is \emph{more} than the sum of its parts.
\end{thm}
```

you'll get

**Theorem 2.1.** *The whole is* more *than the sum of its parts.*

*Note.* The `\emph` command is a recent addition to LaTeX, and it has the feature that it automatically inserts an italic correction where needed. If you don't know what an italic correction is, you can safely ignore this paragraph (but I will at least mention that all those "`\/`" commands frequently seen in TeX (and older LaTeX) documents are all inserting italic corrections; the point of this paragraph is that, with the current version of LaTeX, you don't have to do that anymore).

2.5. **Once is never enough.** This is an explanation of how LaTeX manages to fill in cross-references to parts of the file it hasn't processed yet, and what those `.aux` and `.toc` files are.

*Cross-References.* Every time LaTeX processes your file, it writes an *auxiliary* file. Since the file containing these instructions is called `amshelp.tex`, the auxiliary file is called `amshelp.aux`. The auxiliary file contains the definitions of all the keys used for cross-references. When LaTeX begins to process your file, it first looks for an `.aux` file, and reads it in if it exists. Of course, this is the `.aux` file that was produced the *last* time that your file was processed, so the Theorem numbers, Section numbers, etc., are all the ones from the last time the file was processed.

The very first time that LaTeX processes your file, there is no `.aux` file, and so LaTeX gives *lots* of warning messages about undefined labels, or whatever. Ignore all of this. The *next* time that you run LaTeX, there *will* be an `.aux` file, and all the references will be filled in. (Yes, it is possible, at least in theory, for some page number to change every time you run LaTeX on your file, even without any changes in the source file, but this isn't very likely.)

*The Table of Contents.* If you give the command `\tableofcontents`, then LaTeX will try to write a table of contents, including the page numbers of the sections. Obviously, LaTeX can't know those page numbers or section titles yet, so as LaTeX processes your file, it writes a `.toc` file containing the information it needs. (The `.toc` file for these instructions is `amshelp.toc`.) Once again, LaTeX is always using the information from the *last* time that it processed your file.

If you *do* include a table of contents in your document, and if the table of contents takes up at least a page or so of space, then you might have to run LaTeX *three* times in order to get all of the cross-references right. The reason for this is that the first time you run LaTeX there isn't any `.toc` file listing the section titles, and so the table of contents has nothing in it. The second time you run LaTeX you'll get a table of contents that lists the page numbers for the sections from the last time you ran LaTeX, when the table of contents took up no space at all. Unfortunately, during this second run, the table of contents will be created, and will take up enough space to change the page numbers of the sections from what they were during the first run. Only during the *third* run will the correct page numbers be written into the table of contents. Since this doesn't change the amount of space that the table of contents occupies, this version will be correct.

*How do I know when everything is correct?* After processing your file, LaTeX checks whether all the cross-reference numbers that it read from the `.aux` file are correct. If any of them are incorrect, it prints a warning on the screen at the very end of the run advising you that labels may have changed and that you should run LaTeX again to get the cross-references right. Unfortunately, LaTeX doesn't seem to check that the table of contents entries are correct, so if you change the name of a section in a way that doesn't make any page references incorrect, you won't be warned to run LaTeX again.

## 3. Title, Author, and the `\maketitle` command

This stuff should go right after the `\begin{document}` command. I'll give a quick sketch here, which is probably all you'll ever need, but the full explanation is given in *Instructions for preparation of papers and monographs: AMS-LaTeX* [2]. If you are already familiar with LaTeX, then you should be warned that this part is slightly different from what you do when using the standard LaTeX `article` document class.

3.1. **The title.** You specify the title with the command

```
\title[Optional running title]{Actual title}
```

These instructions used the title command

```
\title[Running \AmS-\LaTeX]{Getting up and running\\
                            with \AmS-\LaTeX}
```

Notice that you indicate line breaks in the title with a double backslash. If I had decided to omit the line break and have the full title printed in the head of the odd numbered pages, I would have used the command

```
\title{Getting up and running with \AmS-\LaTeX}
```

3.2. **The author, and the author's address.** The author is specified with an `author` command:

```
\author{Author's name}
```

These directions used the command `\author{Philip S. Hirschhorn}`. The author's address is given in an address command, with double backslashes to indicate line breaks. These instructions used the command

```
\address{Department of Mathematics\\
Wellesley College\\
Wellesley, Massachusetts 02481}
```

If the author's current address is different from the address at which the research was carried out, then you can specify the current address with the command `\curraddr`. For example, you might say

```
\curraddr{Department of Mechanics\\
Brake and Wheel Bearing Division\\
Serene Service Center\\
Salem, Massachusetts 02139}
```

You can also include an email address, with the `\email` command. These instructions used the command

```
\email{psh@math.mit.edu}
```

To acknowledge support, use the command `\thanks`, e.g.,

```
\thanks{Supported in part by NSF grant 3.14159}
```

This will be printed as a footnote on the first page.

*Multiple authors.* If there are several authors, then each one should have a separate `\author` command, with each individual's address, current address, email address, and thanks following that individual's `\author` command, in its own `\address` command (and `\curraddr` command, and `\thanks` command, and `\email` command). If there *are* several authors, and their combined names are too long for the running head on the even numbered pages, you can give an optional argument to each `\author` command to supply a shortened form to use in the running head, as in

```
\author[P.S. Hirschhorn]{Philip S. Hirschhorn}.
```

(It's apparently a convention that the running head in a multiple author paper should have only initials for the first and middle names, but I don't think that I was invited to that convention.)

3.3. **The date.** This is pretty straightforward:

```
\date{Whatever date you please}
```

To have the date of processing used, use the command `\date{\today}`.

3.4. **`\maketitle`.** After you've given all of the commands mentioned in this section, you can give the command `\maketitle`. If you *don't* give the command `\maketitle`, a title won't be made. The exact arrangement of all this information is determined by the document class. In particular, the `amsart` document class puts the author's address at the *end* of the paper.

## 4. Theorems, Propositions, Lemmas, etc.

The instructions in this section assume that you're using the `\newtheorem` commands that I put in the file `template.tex`.

4.1. **Stating theorems, propositions, etc.** To state a theorem, you do the following:

```
\begin{thm}
  The square of the hypotenuse of a right triangle is equal to the
  sum of the squares of the two adjacent sides.
\end{thm}
```

If you do that, you'll get the following:

**Theorem 4.1.** *The square of the hypotenuse of a right triangle is equal to the sum of the squares of the two adjacent sides.*

If you thought that it was only a proposition, you'd use

```
\begin{prop}
  The square of the hypotenuse of a right triangle is equal to the
  sum of the squares of the two adjacent sides.
\end{prop}
```

and you'd get

**Proposition 4.2.** *The square of the hypotenuse of a right triangle is equal to the sum of the squares of the two adjacent sides.*

If you think it's a theorem again, but you'd like to make reference to it in some other part of the paper, you have to choose a *key* with which you'll refer to it, and then *label* the theorem. If you want to use the key *pythagthm*, then it would look like the following:

```
\begin{thm}
  \label{pythagthm}
  The square of the hypotenuse of a right triangle is equal to the
  sum of the squares of the two adjacent sides.
\end{thm}
```

If you later give the command `\ref{pythagthm}`, then that command will expand to the *number* that was assigned to that theorem (in this case, 4.1). For more explanation of cross-references, see section 5.

If you'd like to state a theorem and give a *name* to it, then you can add an optional argument to the `\begin{thm}` command. If you type

```
\begin{thm}[Pythagoras]
  The square of the hypotenuse of a right triangle is equal to the
  sum of the squares of the two adjacent sides.
\end{thm}
```

you'll get

**Theorem 4.3** (Pythagoras)**.** *The square of the hypotenuse of a right triangle is equal to the sum of the squares of the two adjacent sides.*

*Summary of environments provided in the template.* All of the following structures are numbered in the same sequence, in the form SectionNumber.Number. Equations (i.e., displayed formulas, whether they are equations or not) will be numbered in the same sequence.

Theorem Environments

| Name | Printed Form | Body font |
|---|---|---|
| `thm` | **Theorem** | Italic |
| `cor` | **Corollary** | Italic |
| `lem` | **Lemma** | Italic |
| `prop` | **Proposition** | Italic |
| `defn` | **Definition** | Normal |
| `rem` | *Remark* | Normal |
| `ex` | *Example* | Normal |
| `notation` | *Notation* | Normal |
| `terminology` | *Terminology* | Normal |

For full details, see the beginning of the template file (reproduced here in section 12), after the comment "The Theorem Environments."

4.2. **Proofs.** To give a proof, you do the following:

```
\begin{proof}
  As any fool can plainly see, it's true!
\end{proof}
```

and you'll get the following:

*Proof.* As any fool can plainly see, it's true! □

If the theorem said that a condition was both necessary and sufficient for something, and you want to prove each part separately, you can do the following:

```
\begin{proof}[Proof (sufficiency)]
  Well, it's \emph{obviously} sufficient!
\end{proof}
```

and you'll get

*Proof (sufficiency).* Well, it's *obviously* sufficient! □

that is, the `proof` environment allows you to use an optional second argument that will appear in place of the word `Proof`.

If the proof of Theorem 4.1 does not appear immediately after its statement, you might use the following:

```
\begin{proof}[Proof of Theorem~\ref{pythagthm}]
  As any fool can plainly see, it's true!
\end{proof}
```

and you'd get

*Proof of Theorem 4.1.* As any fool can plainly see, it's true! □

## 5. Cross-References

This section explains how to make reference to numbered sections, theorems, equations, and bibliography items, with the correct reference numbers filled in automatically by LaTeX.

5.1. **References to sections, theorems and equations.** For each structure in the manuscript to which you'll be making reference, you must assign a *key* that you'll use to refer to that structure. For sections, theorems and numbered equations, you assign the key using the `\label` command. This command takes one argument, which is the *key* you're assigning to the object. The command `\ref{key}` then produces the number that was assigned to that structure. If the structure is an equation, then the command `\eqref{key}` should be used instead of the command `\ref{key}`, since `\eqref{key}` provides parentheses around the equation number.

Consider the following example.

**Theorem 5.1.** *If the maps $f\colon X \to Y$ and $g\colon X \to Y$ are homotopic, then the induced homomorphisms $f_*\colon \mathrm{H}_*X \to \mathrm{H}_*Y$ and $g_*\colon \mathrm{H}_*X \to \mathrm{H}_*Y$ are equal.*

We typed that theorem as follows.

```
\begin{thm}
  \label{homotopy}
  If the maps $f\colon X \to Y$ and $g\colon X \to Y$ are homotopic,
  then the induced homomorphisms $f_{*} \colon \mathrm{H}_{*}X \to
  \mathrm{H}_{*}Y$ and $g_{*} \colon \mathrm{H}_{*}X \to
  \mathrm{H}_{*}Y$ are equal.
\end{thm}
```

If we now type "`see Theorem~\ref{homotopy}`," then it will be printed as "see Theorem 5.1."

*So, what exactly is the label labeling?* We began this section by typing

```
\section{Cross-References}
\label{sec:xreferences}
```

and we began this subsection by typing

```
\subsection{References to sections, theorems and equations}
\label{sec:thmrefs}
```

The phrase "`See section~\ref{sec:xreferences}`" is printed as "See section 5," while the phrase "`See section~\ref{sec:thmrefs}`" is printed as "See section 5.1."

The command `\label{key}` assigns to `key` the value of the *smallest enclosing structure*. That's why the command `\ref{sec:xreferences}` is printed as 5, while `\ref{sec:thmrefs}` is printed as 5.1: the key `sec:xreferences` was defined inside of section 5 but *outside* of section 5.1, while the key `sec:thmrefs` was defined *inside* of section 5.1.

*References to equations.* To make reference to a numbered equation, you assign the *key* as before, but you replace `\ref` with `\eqref`, so that parentheses will be printed around the equation number. For example, if you type

```
\begin{equation}
  \label{additivity}
  \mathrm{H}_{*} \bigvee_{\alpha\in A} X_{\alpha}   \approx
    \bigoplus_{\alpha\in A}\mathrm{H}_{*} X_{\alpha}
\end{equation}
```

then you'll get

$$\mathrm{H}_{*} \bigvee_{\alpha\in A} X_{\alpha} \approx \bigoplus_{\alpha\in A}\mathrm{H}_{*} X_{\alpha} \tag{5.2}$$

If we now say

```
\begin{thm}
  Equation~\eqref{additivity} is true for all sorts of functors
  $\mathrm{H}$.
\end{thm}
```

then we'll get

**Theorem 5.3.** *Equation* (5.2) *is true for all sorts of functors* H.

Notice the parentheses around the equation number, and the fact that even though the theorem is set in slanted type, the equation number is set in an upright font? This is the difference between `\eqref` and `\ref`; the command `\eqref` arranges it so that the number and surrounding parentheses are in an upright font no matter what the surrounding font, and supplies an italic correction if it's needed.

5.2. **References to page numbers.** If you want to make reference to the *page* that contains a label, rather than to the structure that is labeled, use the command `\pageref{key}`. For example, if you type

```
See page~\pageref{homotopy} to find Theorem~\ref{homotopy}.
```

you'll get "See page 9 to find Theorem 5.1."

5.3. **Bibliographic references.** Bibliography entries receive a *key* as part of their basic structure. Each item in the bibliography is entered as

```
\bibitem{key} The actual bibliography item goes here.
```

(For more detail on this, see section 11.)

You refer to bibliography items using the `\cite` command. For example, the bibliography of these instructions contains the entry

```
\bibitem{HA}
D. G. Quillen, \emph{Homotopical Algebra,} Lecture Notes in
Mathematics number 43, Springer-Verlag, Berlin, 1967.
```

If we say "`This is the work of Quillen~\cite{HA}`," then it will be printed as "This is the work of Quillen [8]." Notice that square brackets have been inserted around the bibliography item number.

The `\cite` command takes an optional argument, which allows you to annotate the reference. If we say "`see~\cite[Chapter I]{HA}`", then it will be printed as "see [8, Chapter I]".

## 6. Mathematics in running text

This is pretty much exactly as it is in plain TeX, except that you have an extra option (which you can ignore). The simplest thing is to just enclose between dollar signs any material that should be in math mode. Thus, if you type

```
Let $f\colon X \to Y$ be a continuous function.
```

you'll get

Let $f\colon X \to Y$ be a continuous function.

The only novelty that LaTeX introduces is that, instead of using a dollar sign to toggle math mode on and off, you can use '`\(`' to *begin* math mode, and '`\)`' to *end* math mode. Thus, the example above is equivalent to typing

```
Let \(f\colon X \to Y\) be a continuous function.
```

This provides a tiny bit more error checking, but can otherwise be safely ignored.

## 7. Displayed mathematics

7.1. **Single line displays.** To display mathematics and number the display (so that you can refer to it from elsewhere in the paper), you use the `equation` environment. (LaTeX calls all such numbers *equation numbers*, whether or not the display has anything to do with equations.) If you type

```
\begin{equation}
  \pi_{1}(X \vee Y) \approx \pi_{1}X * \pi_{1}Y
\end{equation}
```

you'll get

$$\pi_1(X \vee Y) \approx \pi_1 X * \pi_1 Y \tag{7.1}$$

If you'd like to be able to make reference to the equation number, you need to *label* the equation, using a *key* that you can use for referencing it:

```
\begin{equation}
  \label{pi1eqn}
  \pi_{1}(X \vee Y) \approx \pi_{1}X * \pi_{1}Y
\end{equation}
```

If you later type "`see formula~\eqref{pi1eqn}`" you'll get "see formula (7.1)." (For more on cross-references to formulas, see section 5.1.)

To display mathematics without an equation number, you use the `equation*` environment. (This is a common LaTeXism: Adding an asterisk to the name of a numbered LaTeX environment often gives the unnumbered equivalent.) If you type

```
\begin{equation*}
  \pi_{1}(X \vee Y) \approx \pi_{1}X * \pi_{1}Y
\end{equation*}
```

then you'll get

$$\pi_1(X \vee Y) \approx \pi_1 X * \pi_1 Y$$

There are several essentially equivalent ways of producing this unnumbered display: The `displaymath` environment, used as in

```
\begin{displaymath}
  \pi_{1}(X \vee Y) \approx \pi_{1}X * \pi_{1}Y
\end{displaymath}
```

accomplishes the same thing, as will either

```
$$
  \pi_{1}(X \vee Y) \approx \pi_{1}X * \pi_{1}Y
$$
```

or

```
\[
  \pi_{1}(X \vee Y) \approx \pi_{1}X * \pi_{1}Y
\]
```

It's generally thought to be a better idea to use either the `equation*` environment or the `displaymath` environment for displays, and to avoid the double dollar signs and the `\[`, `\]` pair.

7.2. **Displaying several equations without alignment.** You can put several displayed lines together, each one centered with no alignment between the different lines, using the `gather` environment. When typing this, the lines are separated by a double backslash `\\`. For example, if you type

```
\begin{gather}
  (X\otimes L) \amalg_{(X\otimes K)} (Y\otimes K)
    \longrightarrow Y\otimes L\\
  X^{L} \longrightarrow X^{K} \times_{Y^{K}} Y^{L}
\end{gather}
```

then you'll get

$$(X\otimes L) \amalg_{(X\otimes K)} (Y\otimes K) \longrightarrow Y\otimes L \tag{7.2}$$

$$X^{L} \longrightarrow X^{K} \times_{Y^{K}} Y^{L} \tag{7.3}$$

The `gather*` environment would produce the same thing without the equation numbers. You can also label each line so that you can refer to them: If you had typed that as

```
\begin{gather}
  \label{eq:push}
  (X\otimes L) \amalg_{(X\otimes K)} (Y\otimes K)
    \longrightarrow Y\otimes L\\
  \label{eq:pull}
  X^{L} \longrightarrow X^{K} \times_{Y^{K}} Y^{L}
\end{gather}
```

and then typed "`see \eqref{eq:push} or \eqref{eq:pull}`", you'd get "see (7.2) or (7.3)".

7.3. **Displays over several lines without alignment.** For a long display that must be broken across several lines, you can use the `multline` environment. When typing this, the lines are separated by a double backslash `\\`. The first line will be shifted left of center, the last will be shifted right of center, and the lines in between those will be centered. For example, if you type

```
\begin{multline}
  \label{eq:BigComp}
  \mathrm{F} X\otimes\Delta[n]   \xrightarrow{1 \otimes D}
    \mathrm{F} X\otimes(\Delta[n]\times\Delta[n])\\
  \xrightarrow{\sigma}
    \mathrm{F}\bigl(X\otimes(\Delta[n]\times\Delta[n])\bigr)
    \approx \mathrm{F}\bigl((X\otimes\Delta[n])\otimes\Delta[n]\bigr)\\
  \xrightarrow{\mathrm{F}(\alpha\otimes 1)}
    \mathrm{F}(Y\otimes\Delta[n]) \xrightarrow{\mathrm{F}(\beta)}
    \mathrm{F}(Z)
\end{multline}
```

then you'll get

$$\begin{multline}\mathrm{F} X\otimes\Delta[n] \xrightarrow{1 \otimes D} \mathrm{F} X\otimes(\Delta[n]\times\Delta[n])\\ \xrightarrow{\sigma} \mathrm{F}\bigl(X\otimes(\Delta[n]\times\Delta[n])\bigr) \approx \mathrm{F}\bigl((X\otimes\Delta[n])\otimes\Delta[n]\bigr)\\ \xrightarrow{\mathrm{F}(\alpha\otimes 1)} \mathrm{F}(Y\otimes\Delta[n]) \xrightarrow{\mathrm{F}(\beta)} \mathrm{F}(Z)\end{multline} \tag{7.4}$$

You can then type "`the composition \eqref{eq:BigComp}`" and it will appear as "the composition (7.4)".

The `multline*` environment is similar, except that it omits the equation number.

7.4. **Displays with alignment.** To display several lines of mathematics with alignment between the lines, you use the `align` environment. When typing this, the lines are separated by a double backslash `\\` and each line has an ampersand `&` immediately preceding the symbol to be aligned with the corresponding symbols on the other lines. For example, if you type

```
\begin{align}
  \label{eq:pi1}
  \pi_{1}(X\vee Y) &\approx \pi_{1}X * \pi_{1}Y\\
  \label{eq:additivity}
  \widetilde{\mathrm{H}}_{*}(X\vee Y) &\approx
    \widetilde{\mathrm{H}}_{*}X \oplus \widetilde{\mathrm{H}}_{*}Y
\end{align}
```

then you'll get

$$\pi_1(X \vee Y) \approx \pi_1 X * \pi_1 Y \tag{7.5}$$

$$\widetilde{\mathrm{H}}_*(X \vee Y) \approx \widetilde{\mathrm{H}}_* X \oplus \widetilde{\mathrm{H}}_* Y \tag{7.6}$$

and if you type "`see \eqref{eq:pi1} or \eqref{eq:additivity}`" then you'll get "see (7.5) or (7.6)".

The `align*` environment is similar, except that it omits the equation numbers. For example, if you type

```
\begin{align*}
  \mathcal{M}\bigl((\operatorname{colim} \boldsymbol{X})
    \otimes K,Y\bigr) &\approx
    \mathcal{M}(\operatorname{colim} \boldsymbol{X},Y^K)\\
  &\approx \lim \mathcal{M}(\boldsymbol{X},Y^K)\\
  &\approx \lim \mathcal{M}(\boldsymbol{X} \otimes K,Y)\\
  &\approx \mathcal{M}\bigl(\operatorname{colim}
    (\boldsymbol{X} \otimes K),Y\bigr)
\end{align*}
```

then you'll get

$$\begin{aligned}
\mathcal{M}\bigl((\operatorname{colim} \boldsymbol{X}) \otimes K,Y\bigr) &\approx \mathcal{M}(\operatorname{colim} \boldsymbol{X},Y^K)\\
&\approx \lim \mathcal{M}(\boldsymbol{X},Y^K)\\
&\approx \lim \mathcal{M}(\boldsymbol{X} \otimes K,Y)\\
&\approx \mathcal{M}\bigl(\operatorname{colim}(\boldsymbol{X} \otimes K),Y\bigr)
\end{aligned}$$

7.4.1. *Multiple alignment points.* The `align` and `align*` environments can be used to put multiple displays on each of multiple lines, with alignment between the lines. When typing this, the lines are separated by a double backslash `\\`, the different displays on each line are separated by an ampersand `&`, and the symbols to be aligned are preceded by an ampersand `&`. For example, if you type

```
\begin{align*}
  K &\approx G * H&   i&= j+k&  B &\subset C\\
  H &\approx A_{0}*B_{0}&  i'&= j'+k'&  C &= D\cap E\\
  G &\approx \coprod_{\alpha\in A} L_{\alpha}&  i''&=j''+k''&
                                              A &= D \cup E
\end{align*}
```

then you'll get

$$K \approx G * H \qquad i = j + k \qquad B \subset C$$
$$H \approx A_0 * B_0 \qquad i' = j' + k' \qquad C = D \cap E$$
$$G \approx \coprod_{\alpha \in A} L_\alpha \qquad i'' = j'' + k'' \qquad A = D \cup E$$

For another example, if you type

```
\begin{align*}
  \pi_{1}(X\vee Y) &\approx \pi_{1}(X) * \pi_{1}(Y)
    &&\text{(by the van Kampen theorem)}\\
  &\approx G*H &&\text{(by the computation in the previous section)}
\end{align*}
```

then you'll get

$$\begin{aligned} \pi_1(X \vee Y) &\approx \pi_1(X) * \pi_1(Y) && \text{(by the van Kampen theorem)} \\ &\approx G * H && \text{(by the computation in the previous section)} \end{aligned}$$

If you want to specify the separation between the columns in the alignment, you should use the `alignat` environment (see section 7.4.3).

7.4.2. *Alignments flush left and flush right.* To produce alignments similar to those in section 7.4.1 except with the leftmost column flush left and the rightmost column flush right, you use the `flalign` environment (or, to omit the equation numbers, the `flalign*` environment). For example, if you type

```
\begin{flalign*}
  K &\approx G * H&   i&= j+k&  B &\subset C\\
  H &\approx A_{0}*B_{0}&  i'&= j'+k'&  C &= D\cap E\\
  G &\approx \coprod_{\alpha\in A} L_{\alpha}&  i''&=j''+k''&
                                            A &= D \cup E
\end{flalign*}
```

then you'll get

$$K \approx G * H \qquad i = j + k \qquad B \subset C$$
$$H \approx A_0 * B_0 \qquad i' = j' + k' \qquad C = D \cap E$$
$$G \approx \coprod_{\alpha \in A} L_\alpha \qquad i'' = j'' + k'' \qquad A = D \cup E$$

7.4.3. *Multiple alignment points with chosen spacing.* To produce alignments as in section 7.4.1 except with the ability to choose the horizontal space between the columns, you use the `alignat` environment (or, to omit the equation numbers, the `alignat*` environment). These environments don't insert any horizontal space between the columns, and so you can insert the exact amount of space you want by including it at the beginning of one of the columns.

The format of `alignat` is slightly different from that of the `align` environment in that you must include an argument specifying the number of columns. For example, if you type

```
\begin{alignat}{2}
  K &\approx G*H&  \quad&\text{(by an earlier theorem)}\\
  A &\approx \lim_{i\in I} A_{i}&  &\text{(by the definition of $A$)}
```

```
\end{alignat}
```

then you'll get

$$K \approx G * H \quad \text{(by an earlier theorem)} \tag{7.7}$$

$$A \approx \lim_{i\in I} A_i \quad \text{(by the definition of $A$)} \tag{7.8}$$

Note that it's only necessary to insert the space in one row; the alignment forces the space to appear in all rows.

7.5. **Displays that are part of a larger display.** In addition to the environments that create an entire display (`equation`, `multline`, `align`, `flalign`, and `alignat`), there are environments that can be used as components of a larger display:

| | | |
|---|---|---|
| `gathered` | is similar to | `gather*` |
| `aligned` | is similar to | `align*` |
| `alignedat` | is similar to | `aligned*` |

(There is also a `split` environment, which is similar to the `aligned` environment, and a `cases` environment, described in section 7.6.)

For example, if you type

```
\begin{equation}
  \begin{aligned}
    A &= B+C\\
    &= D+E
  \end{aligned}
  \qquad \text{and, in addition,}\qquad
  \begin{gathered}
    x^{2} + y^{2} = z^{2}\\
    a=b+c
  \end{gathered}
\end{equation}
```

then you'll get

$$\begin{aligned} A &= B+C\\ &= D+E \end{aligned} \qquad \text{and, in addition,}\qquad \begin{gathered} x^{2} + y^{2} = z^{2}\\ a=b+c \end{gathered} \tag{7.9}$$

7.6. **Cases.** There is a `cases` environment, which constructs the usual display of several cases, and which is used as a part of one of the displayed mathematics environments. For example, if you type

```
\begin{equation}
  \label{eq:abs}
  |x| =
  \begin{cases}
    x& \text{if $x \ge 0$}\\
    -x& \text{if $x < 0$}
  \end{cases}
\end{equation}
```

then you'll get

$$|x| = \begin{cases} x & \text{if $x \ge 0$}\\ -x & \text{if $x < 0$} \end{cases} \tag{7.10}$$

For another example, if you type

```
\begin{align*}
  d_{i} \sigma &=
  \begin{cases}
    \alpha_{1} \xrightarrow{\sigma_{1}} \alpha_{2}
          \xrightarrow{\sigma_{2}} \cdots
          \xrightarrow{\sigma_{n-1}} \alpha_{n}
      &\text{if $i=0$}\\
    \alpha_0 \xrightarrow{\sigma_{0}} \cdots
          \xrightarrow{\sigma_{i-2}} \alpha_{i-1}
          \xrightarrow{\sigma_{i}\sigma_{i-1}} \alpha_{i+1}
          \xrightarrow{\sigma_{i+1}} \cdots
          \xrightarrow{\sigma_{n-1}} \alpha_{n}
      &\text{if $0<i<n$}\\
    \alpha_{0} \xrightarrow{\sigma_{0}} \alpha_{1}
          \xrightarrow{\sigma_1} \cdots
          \xrightarrow{\sigma_{n-2}} \alpha_{n-1}
      &\text{if $i=n$}
  \end{cases}\\
  s_{i} \sigma &= \alpha_{0} \xrightarrow{\sigma_{0}} \cdots
    \xrightarrow{\sigma_{i-1}} \alpha_{i}
    \xrightarrow{1_{\alpha_{i}}} \alpha_{i}
    \xrightarrow{\sigma_{i}} \alpha_{i+1}
    \xrightarrow{\sigma_{i+1}} \cdots
    \xrightarrow{\sigma_{n-1}} \alpha_{n}
\end{align*}
```

then you'll get

$$\begin{aligned}
  d_{i} \sigma &=
  \begin{cases}
    \alpha_{1} \xrightarrow{\sigma_{1}} \alpha_{2}
          \xrightarrow{\sigma_{2}} \cdots
          \xrightarrow{\sigma_{n-1}} \alpha_{n}
      &\text{if $i=0$}\\
    \alpha_0 \xrightarrow{\sigma_{0}} \cdots
          \xrightarrow{\sigma_{i-2}} \alpha_{i-1}
          \xrightarrow{\sigma_{i}\sigma_{i-1}} \alpha_{i+1}
          \xrightarrow{\sigma_{i+1}} \cdots
          \xrightarrow{\sigma_{n-1}} \alpha_{n}
      &\text{if $0<i<n$}\\
    \alpha_{0} \xrightarrow{\sigma_{0}} \alpha_{1}
          \xrightarrow{\sigma_1} \cdots
          \xrightarrow{\sigma_{n-2}} \alpha_{n-1}
      &\text{if $i=n$}
  \end{cases}\\
  s_{i} \sigma &= \alpha_{0} \xrightarrow{\sigma_{0}} \cdots
    \xrightarrow{\sigma_{i-1}} \alpha_{i}
    \xrightarrow{1_{\alpha_{i}}} \alpha_{i}
    \xrightarrow{\sigma_{i}} \alpha_{i+1}
    \xrightarrow{\sigma_{i+1}} \cdots
    \xrightarrow{\sigma_{n-1}} \alpha_{n}
\end{aligned}$$

## 8. Commutative diagrams using XY-pic

XY-pic is a powerful package that enables you to draw very complex commutative diagrams within your LaTeX file, avoiding the need to create the graphics separately and then import them. XY-pic isn't a part of $\mathcal{A}_\mathcal{M}\mathcal{S}$-LaTeX, but XY-pic and $\mathcal{A}_\mathcal{M}\mathcal{S}$-LaTeX work quite well together. We'll describe the most important features of XY-pic's commutative diagram macros here. The full documentation for commutative diagrams can be found in the *XY-pic User's Guide* [9], and the full documentation for all of XY-pic can be found in the *XY-pic Reference Manual* [10].

To use XY-pic to draw commutative diagrams, you include the lines

```
\usepackage[all,cmtip]{xy}
\let\objectstyle=\displaystyle
```

in the preamble of your document, i.e., after the `\documentclass` command and before the `\begin{document}` command. You can omit the second of those two lines if you want the nodes of your diagram to be in `\textstyle` by default.

You create a diagram using the `\xymatrix` command, immediately followed by a pair of braces that enclose the diagram specification. The diagram is described in a manner similar to the way matrices are described:

(1) There is a rectangular array of nodes of the digram.
(2) The nodes in a row are separated by an ampersand `&`, and the rows are separated by a double backslash `\\`.
(3) An arrow is specified immediately following the node that is its source. A straight line arrow is specified by `\ar[target]`, where `target` is
   - `r` for the node one place to the right,
   - `l` for the node one place to the left,
   - `d` for the node one place down,
   - `u` for the node one place up,
   - `rr` for the node two places to the right,
   - `dr` for the node one place down and one place to the right,
   - etc.

It's also possible to have labels on the arrows (see section 8.3), arrows that are built of things other than a single solid line with a single arrowhead at the end (see section 8.4), curved arrows (see section 8.5), and arrows that pass over or under other arrows (see section 8.2).

For example, if you type

```
\begin{equation}
  \xymatrix{
    {A}\ar[r] \ar[d]
    &{B} \ar[d]\\
    {C} \ar[r]
    &{D}
  }
\end{equation}
```

then you'll get

$$\begin{tikzcd} A \arrow[r] \arrow[d] & B \arrow[d] \\ C \arrow[r] & D \end{tikzcd} \tag{8.1}$$

If you'd like the equation number to be centered vertically with respect to the diagram, you should enclose the `\xymatrix` command in a `\vcenter`, as in

```
\begin{equation}
  \vcenter{
    \xymatrix{
      {A}\ar[r] \ar[d]
      &{B} \ar[d]\\
      {C} \ar[r]
      &{D}
    }
  }
\end{equation}
```

in which case you'll get

$$\begin{tikzcd} A \arrow[r] \arrow[d] & B \arrow[d] \\ C \arrow[r] & D \end{tikzcd} \tag{8.2}$$

Not all nodes need appear. To omit a node in the middle of a line, just type the `&` that moves you on to the next node. To omit a node at the end of the line, just type the line-ending `\\` and go on to the next line. For example, if you type

```
\begin{equation*}
  \xymatrix{
    {A} \ar[rr] \ar[dr] \ar[d]
    && {B} \ar[dd]\\
    {C} \ar[d] \ar[drr]
    & {D} \ar[l] \ar[ur]\\
    {E} \ar[rr]
    &&{F}
  }
\end{equation*}
```

you'll get

$$\begin{tikzcd} A \arrow[rr] \arrow[dr] \arrow[d] & & B \arrow[dd] \\ C \arrow[d] \arrow[drr] & D \arrow[l] \arrow[ur] & \\ E \arrow[rr] & & F \end{tikzcd}$$

8.1. **Changing the spacing.** You can change the space between the rows of a diagram and the space between the columns of a diagram. The following commands can be inserted following the command `\xymatrix` and before the left brace that begins the diagram specification:

| | |
|---|---|
| `@=dimen` | set row and column spacing to `dimen` |
| `@R=dimen` | set row spacing to `dimen` |
| `@C=dimen` | set column spacing to `dimen` |

The `dimen` in all of these commands can be positive, zero, or negative. (Negative `dimen`'s can be useful in diagrams with very wide entries; see (8.3).)

For example, if you type

```
\begin{displaymath}
  \vcenter{
    \xymatrix@=2ex{
      {A} \ar[r] \ar[d]
      & {B} \ar[d]\\
      {C} \ar[r]
      & {D}
    }% \xymatrix
  }% \vcenter
  \text{\qquad or\qquad}
```

```
  \vcenter{
    \xymatrix{
      {A} \ar[r] \ar[d]
      & {B} \ar[d]\\
      {C} \ar[r]
      & {D}
    }% \xymatrix
  }% \vcenter
  \text{\qquad or\qquad}
  \vcenter{
    \xymatrix@C=10ex{
      {A} \ar[r] \ar[d]
      & {B} \ar[d]\\
      {C} \ar[r]
      & {D}
    }% \xymatrix
  }% \vcenter
\end{displaymath}
```

then you'll get

$$\begin{array}{ccc} A & \to & B \\ \downarrow & & \downarrow \\ C & \to & D \end{array} \qquad \text{or} \qquad \begin{array}{ccc} A & \longrightarrow & B \\ \Big\downarrow & & \Big\downarrow \\ C & \longrightarrow & D \end{array} \qquad \text{or} \qquad \begin{array}{ccc} A & \xrightarrow{\hspace{5em}} & B \\ \Big\downarrow & & \Big\downarrow \\ C & \xrightarrow{\hspace{5em}} & D \end{array}$$

For an example of the usefulness of negative column spacing: If you type

```
\newcommand{\pushout}[3]{#1\mathbin{\mathord{\amalg}_{#2}}#3}
\begin{equation}
  \xymatrix@C=-12em{
    {\pushout{(A\otimes L)}
             {A\otimes K}
             {(B\otimes K)}
           } \ar[r] \ar[dr]
    & {B\otimes L}
    & {\pushout{(P\otimes L)}
               {P\otimes M}
               {(Q\otimes M)}
           } \ar[l] \ar[dl]\\
    & {\pushout{\bigl(\pushout{(A\otimes L)}
                              {A\otimes K}
                              {(B\otimes K)}\bigr)}
               {\pushout{(X\otimes L)}
                        {X\otimes K}
                        {(Y\otimes K)}}
               {\bigl(\pushout{(P\otimes L)}
                              {P\otimes M}
                              {(Q\otimes M)}\bigr)}
           } \ar[u]
  }
```

```
\end{equation}
```

then you'll get

$$\begin{tikzcd}[column sep=small]
(A \otimes L) \amalg_{A \otimes K} (B \otimes K) \arrow[r] \arrow[dr] & B \otimes L & (P \otimes L) \amalg_{P \otimes M} (Q \otimes M) \arrow[l] \arrow[dl] \\
& \big((A \otimes L) \amalg_{A \otimes K} (B \otimes K)\big) \amalg_{(X \otimes L) \amalg_{X \otimes K} (Y \otimes K)} \big((P \otimes L) \amalg_{P \otimes M} (Q \otimes M)\big) \arrow[u] &
\end{tikzcd} \tag{8.3}$$

which would never fit on the page without negative column spacing.

8.2. **Arrows passing under.** Arrows can be created that go to a sequence of nodes, passing under (i.e., leaving a small gap at) all the intermediate nodes. If an arrow is created that passes under an empty node while a second arrow crosses that node in the normal way, the effect is that the first arrow passes under the second arrow.

To draw an arrow that passes under a sequence of nodes and then goes on to a final node, the `\ar` is followed by `'[node]` for each node that you pass under, followed by `[finalnode]`. For example, if you type

```
\begin{displaymath}
  \xymatrix{
    {A} \ar '[r] '[rr] [rrr]
    & {B}
    & {}
    & {D}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd}
A \arrow[r, no head] & B \arrow[r, no head] & {} \arrow[r] & D
\end{tikzcd}$$

For an example of crossing arrows, if you type

```
\begin{displaymath}
  \xymatrix@=3ex{
    {A} \ar[rr] \ar[dd] \ar'[dr][ddrr]
    && {B} \ar[dd] \ar[ddll]\\
    & {}\\
    {C} \ar[rr]
    && {D}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd}[sep=3ex]
A \arrow[rr] \arrow[dd] \arrow[ddrr] & & B \arrow[dd] \arrow[ddll, crossing over] \\
& {} & \\
C \arrow[rr] & & D
\end{tikzcd}$$

Note that that diagram has three rows and three columns, but nothing appears in either the second row or the second column. (For an explanation of the command `@=3ex`, which changes the size of the diagram, see section 8.1.) For a more elaborate example, if you type

```
\begin{displaymath}
  \xymatrix@=2ex{
    {A} \ar[rr] \ar[dr] \ar[dd]
    && {B} \ar[dr] \ar'[d][dd]\\
    & {A'} \ar[rr] \ar[dd]
    && {B'} \ar[dd]\\
    {C} \ar'[r][rr] \ar[dr]
    && {D} \ar[dr]\\
    & {C'} \ar[rr]
    && {D'}
    }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd}[sep=small]
A \arrow[rr] \arrow[dr] \arrow[dd] & & B \arrow[dr] \arrow[dd] & \\
& A' \arrow[rr, crossing over] & & B' \arrow[dd] \\
C \arrow[rr] \arrow[dr] & & D \arrow[dr] & \\
& C' \arrow[rr] \arrow[from=uu, crossing over] & & D'
\end{tikzcd}$$

8.3. **Labeling the arrows.** It's possible to label an arrow, on one or both sides of the arrow. (It's also possible to have the label "break" the arrow; for this, see section 8.3.4.) By default, the label is located halfway from the center of the source to the center of the target. This will often be halfway along the arrow, but not if the source and the target are of different sizes. There are also options to place the label halfway along the arrow (see section 8.3.1) or at an arbitrary point along the arrow (see section 8.3.2).

To put a label above an arrow (where "above" means when the paper is oriented so that the arrow goes left to right), you type `^{thelabel}` either before or after the target. To put a label below an arrow (where "below" means when the paper is oriented so that the arrow goes left to right), you type `_{thelabel}` either before or after the target. Thus, if you type

```
\begin{displaymath}
  \xymatrix{
    {A}\ar[r]^{f} \ar[d]_{g}
    &{B} \ar[d]^{h}\\
    {C} \ar[r]_{k}
    &{D}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, "g"'] & B \arrow[d, "h"] \\
C \arrow[r, "k"'] & D
\end{tikzcd}$$

8.3.1. *Centering the labels on the arrows.* If you type

```
\begin{displaymath}
  \xymatrix{
    {X\cup_{Y}Z} \ar[r]^{f} \ar[d]_{g}
    &{B} \ar[d]^{h}\\
    {C} \ar[r]_{k}
    &{D}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd} X \cup_Y Z \arrow[r, "f"] \arrow[d, swap, "g"] & B \arrow[d, "h"] \\ C \arrow[r, swap, "k"] & D \end{tikzcd}$$

Note that the label $f$ is halfway from the center of the arrow's source to the center of the arrow's target, but it is not centered along the arrow. To have that label centered along the arrow, we insert a `-` immediately following the `^`, so that we type it as

```
\ar[r]^-{f}
```

and we then get

$$\begin{tikzcd} X \cup_Y Z \arrow[r, "f"] \arrow[d, swap, "g"] & B \arrow[d, "h"] \\ C \arrow[r, swap, "k"] & D \end{tikzcd}$$

For another example, if you type

```
\begin{displaymath}
  \xymatrix{
    {A \amalg_{B} C} \ar[r]^{\text{Add}} \ar[r]_{\sigma} \ar[d]
    & {X} \ar[d]^{\psi}\\
    {Y} \ar[r]^{\alpha} \ar[r]
    & {P\times_{Q}R}
  }
\end{displaymath}
```

you'll get

$$\begin{tikzcd} A \amalg_B C \arrow[r, "\text{Add}", "\sigma"'] \arrow[d] & X \arrow[d, "\psi"] \\ Y \arrow[r, "\alpha"] & P \times_Q R \end{tikzcd}$$

but if you type

```
\begin{displaymath}
  \xymatrix{
    {A \amalg_{B} C} \ar[r]^-{\text{Add}} \ar[r]_-{\sigma} \ar[d]
    & {X} \ar[d]\\
    {Y} \ar[r]^-{\alpha} \ar[r]_-{\text{Lifted}}
```

```
    & {P\times_{Q}R}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd} A \amalg_B C \arrow[r, "\text{Add}", "\sigma"'] \arrow[d] & X \arrow[d] \\ Y \arrow[r, "\alpha", "\text{Lifted}"'] & P \times_Q R \end{tikzcd}$$

8.3.2. *Arbitrary placement of labels.* In addition to the possibility of centering a label along an arrow (see section 8.3.1), it is possible to place a label at an arbitrary point along an arrow. If `a` is a number between 0 and 1, then you can place a label `a` of the way from the center of the source to the center of the target by typing `(a)` immediately following the `^` or `_`. For example, if you type

```
\begin{displaymath}
  \xymatrix@C=8em{
    {A} \ar[r]^(.3){f} \ar[d]
    & {B} \ar[d]\\
    {C} \ar[r]^{g} \ar[d]
    & {D} \ar[d]\\
    {E} \ar[r]_(.7){h}
    & {F}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd}[column sep=8em] A \arrow[r, "f"{pos=0.3}] \arrow[d] & B \arrow[d] \\ C \arrow[r, "g"] \arrow[d] & D \arrow[d] \\ E \arrow[r, "h"'{pos=0.7}] & F \end{tikzcd}$$

For a more elaborate example, with labels along segments of segmented arrows, see diagram (8.4).

8.3.3. *Labeling each segment of a segmented arrow.* An arrow that uses the "arrows passing under" feature (see section 8.2) will be in several segments, and a label can be attached to each of the segments. For example, if you type

```
\begin{displaymath}
  \xymatrix{
    {A} \ar '[r]^{f} '[rr]^{g} [rrr]^{h}
    & {B}
    & {}
    & {D}
  }
\end{displaymath}
```

then you'll get

$$A \overset{f}{\longrightarrow\!\!\!\!-} B \overset{g}{\longrightarrow\!\!\!\!-} \quad \xrightarrow{h} D$$

For a more elaborate example, if you type

```
\begin{equation}
  \label{diag:LblCube}
  \xymatrix@=2ex{
    {A} \ar[rr]^{f} \ar[dr] \ar[dd]_{i}
    && {B} \ar[dr] \ar'[d][dd]^(.3){j}\\
    & {A'} \ar[rr]^(.3){f'} \ar[dd]^(.25){i'}
    && {B'} \ar[dd]^{j'}\\
    {C} \ar'[r]^{g}[rr] \ar[dr]
    && {D} \ar[dr]\\
    & {C'} \ar[rr]_{g'}
    && {D'}
  }
\end{equation}
```

then you'll get

(8.4)

$$\begin{tikzcd}
A \arrow[rr, "f"] \arrow[dr] \arrow[dd, "i"'] & & B \arrow[dr] \arrow[dd, "j" near end] & \\
& A' \arrow[rr, "f'" near start] \arrow[dd, "i'" near start] & & B' \arrow[dd, "j'"] \\
C \arrow[rr, "g"] \arrow[dr] & & D \arrow[dr] & \\
& C' \arrow[rr, "g'"'] & & D'
\end{tikzcd}$$

8.3.4. *Breaking an arrow with a label.* Instead of placing a label to the side of an arrow, you can have the label "break" the arrow. For this, you use the vertical bar character `|` in place of either `^` or `_`.

For example, if you type

```
\begin{displaymath}
  \xymatrix{
    {A} \ar[r]|{f}
    & {B} \ar[r]|{g}
    & {C}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd} A \arrow[r, "f" description] & B \arrow[r, "g" description] & C \end{tikzcd}$$

8.4. **Different arrow styles.** It's possible to have arrows with tails, multiple heads, dotted, dashed, or multiple shafts, and any combination of these (see Table 1). You can even omit both the head and the tail, or omit the arrow entirely, which is useful for placing things into the diagram in places outside of the grid of nodes.

All of these arrows can point in whatever direction you choose; we used `[r]` in the table just for readability. For an example of the use of invisible arrows: If you type

| To produce: | Type: |
|---|---|
| $\cdots\!\!\rightarrow$ | `\ar@{.>}[r]` |
| $\dashrightarrow$ | `\ar@{-->}[r]` |
| $\Longrightarrow$ | `\ar@{=>}[r]` |
| $\vdots\!\Rightarrow$ | `\ar@{:>}[r]` |
| $\twoheadrightarrow$ | `\ar@{->>}[r]` |
| $\rightarrowtail$ | `\ar@{>->}[r]` |
| $\longleftrightarrow$ | `\ar@{<->}[r]` |
| $- - \twoheadrightarrow$ | `\ar@{-->>}[r]` |
| $\succ\!- - \rightarrow$ | `\ar@{>-->}[r]` |
| $\dashleftarrow\!\dashrightarrow$ | `\ar@{<-->}[r]` |
| $\Longrightarrow\!\!\!\!\!\!\;\Rightarrow$ | `\ar@{=>>}[r]` |
| $\Longleftrightarrow$ | `\ar@{<=>}[r]` |
| $\vdots\!\Rightarrow\!\!\!\Rightarrow$ | `\ar@{:>>}[r]` |
| $\Leftarrow\!\vdots\!\Rightarrow$ | `\ar@{<:>}[r]` |
| $\longrightarrow$ without head: $-\!\!-\!\!-$ | `\ar@{-}[r]` |
| $\cdots\cdots$ | `\ar@{.}[r]` |
| $=\!=\!=$ | `\ar@{=}[r]` |
| | `\ar@{}[r]` |

Table 1. Arrow Styles

```
\begin{displaymath}
  \xymatrix{
    {A} \ar@{.>}[r] \ar@{.>}[d] \ar@{}[dr]|{=}
    & {B} \ar[d]\\
    {C} \ar[r]
    & {D}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd} A \arrow[r, dotted] \arrow[d, dotted] \arrow[dr, phantom, "="] & B \arrow[d] \\ C \arrow[r] & D \end{tikzcd}$$

8.5. **Curved arrows.** It's possible to have arrows curve, either by specifying the amount that they curve or by specifying the direction in which they leave their source and the direction from which they arrive at their target.

8.5.1. *Specifying the amount of the curve.* To have an arrow curve in the up direction (where "up" means when the paper is oriented so that the arrow goes left to right) by a default amount you follow the `\ar` with `@/^/`; that is, you type `\ar@/^/[target]`. To have it curve in the down direction by a default amount you follow the `\ar` with `@/_/`; that is, you type `\ar@/_/[target]` (or, if you want to label the arrow, `\ar@/_/[target]^{label}`). Thus, if you type

```
\begin{displaymath}
  \xymatrix{
```

```
    {A} \ar@/^/[drr]^{p} \ar@{.>}[dr]|{\exists!} \ar@/_/[ddr]_{q}\\
    & {B} \ar[r] \ar[d]
    & {C} \ar[d]\\
    & {D} \ar[r]
    & {E}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd}
A \arrow[drr, bend left, "p"] \arrow[dr, dotted, "\exists!" description] \arrow[ddr, bend right, "q"'] & & \\
& B \arrow[r] \arrow[d] & C \arrow[d] \\
& D \arrow[r] & E
\end{tikzcd}$$

If you'd like to specify the amount of curve, you can specify a dimension following the ^ or the _, as in `\ar@/^1ex/[target]`. For example, if you type

```
\begin{displaymath}
  \xymatrix{
    {A} \ar@/^3ex/[drr]^{p} \ar@{.>}[dr]|{\exists!} \ar@/_3ex/[ddr]_{q}\\
    & {B} \ar[r] \ar[d]
    & {C} \ar[d]\\
    & {D} \ar[r]
    & {E}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd}
A \arrow[drr, bend left=40, "p"] \arrow[dr, dotted, "\exists!" description] \arrow[ddr, bend right=40, "q"'] & & \\
& B \arrow[r] \arrow[d] & C \arrow[d] \\
& D \arrow[r] & E
\end{tikzcd}$$

and if you type

```
\begin{displaymath}
  \xymatrix{
    {A} \ar@{>->}[r]^{i}
    & {B} \ar@{->>}[r]^{p}
    & {C} \ar@{.>}@/_4ex/[l]_{s}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd}
A \arrow[r, tail, "i"] & B \arrow[r, two heads, "p"] & C \arrow[l, dotted, bend right=60, "s"']
\end{tikzcd}$$

8.5.2. *Specifying the start and end directions.* To specify the start and end directions of a curved arrow you type `\ar@(start,end)[target]` where "`start`" is the direction towards which the arrow begins and "`end`" is the direction from which the arrow ends. For example, if you type

```
\begin{displaymath}
  \xymatrix{
    {A} \ar[dr]|{p} \ar@(r,u)[dr]^{f} \ar@(d,l)[dr]_{g}\\
    & {B}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd} A \arrow[dr, "p" description] \arrow[dr, bend left=60, "f"] \arrow[dr, bend right=60, "g"'] & \\ & B \end{tikzcd}$$

and if you type

```
\begin{displaymath}
  \xymatrix{
    {A} \ar@(ur,ul)[rr]^{1_{A}} \ar[r]_{f}
    & {B} \ar[r]_{g} \ar@(dr,dl)[rr]_{1_{B}}
    & {A} \ar[r]_{f}
    & {B}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd} A \arrow[r, "f"'] \arrow[rr, bend left, "1_A"] & B \arrow[r, "g"'] \arrow[rr, bend right, "1_B"'] & A \arrow[r, "f"'] & B \end{tikzcd}$$

8.6. **Sliding arrows sideways; multiple arrows.** It's possible to slide arrows sideways, so that you can have more than one straight line arrow between a single pair of nodes. To do this, you type `\ar@<distance>` to move the arrow "upwards" by `distance` (a negative distance will move the arrow "downwards").

For example, if you type

```
\begin{displaymath}
  \xymatrix{
    {A} \ar@<1ex>[r]
    & {B} \ar[r]
    & {C} \ar@<-1ex>[r]
    & {D}
  }
\end{displaymath}
```

then you'll get

$$\begin{tikzcd} A \arrow[r, shift left=1ex] & B \arrow[r] & C \arrow[r, shift right=1ex] & D \end{tikzcd}$$

For another example, if you type

```
\begin{displaymath}
  \xymatrix{
    {\coprod_{i \in I} A_{i}\quad}
      \ar@<2ex>[r]^-{\phi} \ar@<+.6ex>[r]_-{\psi}
    & {\quad\coprod_{i \in I} B_{i}}
  }
\end{displaymath}
```

then you'll get

$$\coprod_{i \in I} A_i \quad \underset{\psi}{\overset{\phi}{\rightrightarrows}} \quad \coprod_{i \in I} B_i$$

## 9. Macro definitions, a.k.a. `\newcommand`

LaTeX allows you to use the same `\def` command that you use in plain TeX, but it's considered bad style. Instead, LaTeX has the `\newcommand` and `\renewcommand` commands, which do a little error checking for you. In plain TeX, you might use the command

```
\def\tensor{\otimes}
```

but in LaTeX, the preferred form is

```
\newcommand{\tensor}{\otimes}
```

The advantage of this is that LaTeX will check to see if there already is a command with the name `\tensor`, and give you an error message if there is. If you know that there is a previous definition of `\tensor` but you *want* to override it, then you use the command

```
\renewcommand{\tensor}{\otimes}
```

If you want to use macros with replaceable parameters, the `newcommand` command allows this. For the equivalent of the plain TeX command

```
\def\pushout#1#2#3{#1\cup_{#2}#3}
```

you use the LaTeX command

```
\newcommand{\pushout}[3]{#1\cup_{#2}#3}
```

i.e., the command name is enclosed in braces, and the number of parameters is enclosed in square brackets.

## 10. Lists: `itemize`, `enumerate`, and `description`

There are three list making environments: `itemize`, `enumerate`, and `description`. The `itemize` environment just lists the items with a marker in front of each one. If you type

```
\begin{itemize}
\item This is the first item in the list, which runs on long enough to
  spill over onto a second line.
\item This is the second item in the list, which is a bit shorter.
\item This is the last item.
\end{itemize}
```

then you'll get

- This is the first item in the list, which runs on long enough to spill over onto a second line.

- This is the second item in the list, which is a bit shorter.
- This is the last item.

The `enumerate` environment looks the same, except that the items in the list are numbered. If you type

```
\begin{enumerate}
\item This is the first item in the list, which runs on long enough to
  spill over onto a second line.
\item This is the second item in the list, which is a bit shorter.
\item This is the last item.
\end{enumerate}
```

then you'll get

(1) This is the first item in the list, which runs on long enough to spill over onto a second line.
(2) This is the second item in the list, which is a bit shorter.
(3) This is the last item.

The `description` environment requires an argument for each `\item` command, which will be printed at the beginning of the item. If you type

```
\begin{description}
\item[sedge] A green plant, found in both wetlands and uplands.
  Sedges are often confused with grasses and rushes.
\item[grass] A green plant, found in both wetlands and uplands.
  Grasses are often confused with sedges and rushes.
\item[rush] A green plant, found in both wetlands and uplands.  Rushes
  are often confused with sedges and grasses
\end{description}
```

you'll get

**sedge:** A green plant, found in both wetlands and uplands. Sedges are often confused with grasses and rushes.
**grass:** A green plant, found in both wetlands and uplands. Grasses are often confused with sedges and rushes.
**rush:** A green plant, found in both wetlands and uplands. Rushes are often confused with sedges and grasses

These environments can be inserted within each other, and the `enumerate` environment keeps track of what level it's at, and numbers its items accordingly. If you type

```
\begin{enumerate}
\item I went to the dry cleaners.
\item I went to the supermarket.  I bought
  \begin{enumerate}
  \item bread,
  \item cheese, and
  \item Tabasco sauce.
  \end{enumerate}
\item I went to the bank.
\end{enumerate}
```

you'll get

(1) I went to the dry cleaners.
(2) I went to the supermarket. I bought
  (a) bread,
  (b) cheese, and
  (c) Tabasco sauce.
(3) I went to the bank.

## 11. The bibliography

**11.1. `\begin{thebibliography}` and `\end{thebibliography}`.** The bibliography is begun with the command

```
\begin{thebibliography}{number}
```

where *number* is any number that, when printed, is as wide as the widest number of any item in the bibliography. (The only use made of `number` is that LaTeX assumes that it's width when printed is at least as large as the width of any number of an item in the bibliography.) For example, if the bibliography will contain between 10 and 19 items, you can use `\begin{thebibliography}{10}`.

After listing each item in the bibliography, you end the bibliography with the `\end{thebibliography}` command.

**11.2. Bibliography items.** Each item is begun with a `\bibitem` command. The format is

```
\bibitem{key for cross-references}Item entry
```

For example, the bibliography in these instructions contains the entry

```
\bibitem{yellowmonster}
A. K. Bousfield and D. M. Kan, \emph{Homotopy Limits, Completions
and Localizations,} Lecture Notes in Mathematics number 304,
Springer-Verlag, New York, 1972.
```

The above entry allows you to say

```
Homotopy inverse limits are discussed
in~\cite[Chapter 11]{yellowmonster}.
```

and have it print as "Homotopy inverse limits are discussed in [4, Chapter 11]." For more on this, see section 5.3.

## 12. The template file

The following is the text of the file `template.tex`.

```
%%% template.tex
%%% This is a template for making up an AMS-LaTeX file
%%% Version of November 16, 2008
%%%---------------------------------------------------------
%%% The following command chooses the default 10 point type.
%%% To choose 12 point, change it to
%%% \documentclass[12pt]{amsart}
\documentclass{amsart}

%%% The following command defines the standard names for all of the
%%% special symbols in the AMSfonts package, listed in
%%% http://www.ctan.org/tex-archive/info/symbols/math/symbols.pdf
```

```
\usepackage{amscd}

%%% The following commands allow you to use \Xy-pic to draw
%%% commutative diagrams.  (You can omit the second line if you want
%%% the default style of the nodes to be \textstyle.)
\usepackage[all,cmtip]{xy}
\let\objectstyle=\displaystyle

%%% If you'll be importing any graphics, uncomment the following
%%% line.  (Note: The spelling is correct; the package graphicx.sty is
%%% the updated version of the older graphics.sty.)
% \usepackage{graphicx}



%%% This part of the file (after the \documentclass command,
%%% but before the \begin{document}) is called the ``preamble''.
%%% This is where we put our macro definitions.

%%% Comment out (or delete) any of these that you don't want to use.
\newcommand{\tensor}{\otimes}
\newcommand{\homotopic}{\simeq}
\newcommand{\homeq}{\cong}
\newcommand{\iso}{\approx}

\DeclareMathOperator{\ho}{Ho}
\DeclareMathOperator*{\colim}{colim}

\newcommand{\R}{\mathbb{R}}
\newcommand{\C}{\mathbb{C}}
\newcommand{\Z}{\mathbb{Z}}

\newcommand{\M}{\mathcal{M}}
\newcommand{\W}{\mathcal{W}}

\newcommand{\itilde}{\tilde{\imath}}
\newcommand{\jtilde}{\tilde{\jmath}}
\newcommand{\ihat}{\hat{\imath}}
\newcommand{\jhat}{\hat{\jmath}}



%%%----------------------------------------------------------------------
%%%----------------------------------------------------------------------
%%% The Theorem environments:
%%%
%%%
%%% The following commands set it up so that:
```

```
%%%
%%% All Theorems, Corollaries, Lemmas, Propositions, Definitions,
%%% Remarks, Examples, Notations, and Terminologies  will be numbered
%%% in a single sequence, and the numbering will be within each
%%% section.  Displayed equations will be numbered in the same
%%% sequence.
%%%
%%%
%%% Theorems, Propositions, Lemmas, and Corollaries will have the most
%%% formal typesetting.
%%%
%%% Definitions will have the next level of formality.
%%%
%%% Remarks, Examples, Notations, and Terminologies will be the least
%%% formal.
%%%
%%% Theorem:
%%% \begin{thm}
%%%
%%% \end{thm}
%%%
%%% Corollary:
%%% \begin{cor}
%%%
%%% \end{cor}
%%%
%%% Lemma:
%%% \begin{lem}
%%%
%%% \end{lem}
%%%
%%% Proposition:
%%% \begin{prop}
%%%
%%% \end{prop}
%%%
%%% Definition:
%%% \begin{defn}
%%%
%%% \end{defn}
%%%
%%% Remark:
%%% \begin{rem}
%%%
%%% \end{rem}
%%%
%%% Example:
%%% \begin{ex}
```

```
%%%
%%% \end{ex}
%%%
%%% Notation:
%%% \begin{notation}
%%%
%%% \end{notation}
%%%
%%% Terminology:
%%% \begin{terminology}
%%%
%%% \end{terminology}
%%%
%%%       Theorem environments

% The following causes equations to be numbered within sections
\numberwithin{equation}{section}

% We’ll use the equation counter for all our theorem environments, so
% that everything will be numbered in the same sequence.

%       Theorem environments

\theoremstyle{plain} %% This is the default, anyway
\newtheorem{thm}[equation]{Theorem}
\newtheorem{cor}[equation]{Corollary}
\newtheorem{lem}[equation]{Lemma}
\newtheorem{prop}[equation]{Proposition}

\theoremstyle{definition}
\newtheorem{defn}[equation]{Definition}

\theoremstyle{remark}
\newtheorem{rem}[equation]{Remark}
\newtheorem{ex}[equation]{Example}
\newtheorem{notation}[equation]{Notation}
\newtheorem{terminology}[equation]{Terminology}

%%%-------------------------------------------------------------------
%%%-------------------------------------------------------------------
%%%-------------------------------------------------------------------
%%%-------------------------------------------------------------------
%%%-------------------------------------------------------------------
%%%-------------------------------------------------------------------
%%%-------------------------------------------------------------------
\begin{document}

%%% In the title, use a double backslash "\\" to show a linebreak:
```

```
%%% Use one of the following two forms:
%%% \title{Text of the title}
%%% or
%%% \title[Short form for the running head]{Text of the title}
\title{}


%%% If there are multiple authors, they're described one at a time:
%%% First author: \author{} \address{} \curraddr{} \email{} \thanks{}
%%% Second author: \author{} \address{} \curraddr{} \email{} \thanks{}
%%% Third author: \author{} \address{} \curraddr{} \email{} \thanks{}
\author{}

%%% In the address, show linebreaks with double backslashes:
\address{}

%%% Current address is optional.
% \curraddr{}

%%% Email address is optional.
% \email{}


%%% If there's a second author:
% \author{}
% \address{}
% \curraddr{}
% \email{}


%%% To have the current date inserted, use \date{\today}:
\date{}

%%% To include an abstract, uncomment the following two lines and type
%%% the abstract in between them:
% \begin{abstract}
% \end{abstract}


\maketitle

%%% To include a table of contents, uncomment the following line:
% \tableofcontents


%%%-----------------------------------------------------------------
%%%-----------------------------------------------------------------
%%% Start the body of the paper here!  E.G., maybe use:
```

```
%%% \section{Introduction}
%%% \label{sec:intro}

%%% For a numbered display, use
%%% \begin{equation}
%%%   \label{something}
%%%   The display goes here
%%% \end{equation}
%%% and you can refer to it as \eqref{something}.

%%% For an unnumbered display, use
%%% \begin{equation*}
%%%   The display goes here
%%% \end{equation*}

%%% To import a graphics file, you must have said
%%% \usepackage{graphicx}
%%% in the preamble (i.e., before the \begin{document}).
%%% Putting it into a figure environment enables it to float to the
%%% next page if there isn't enough room for it on the current page.
%%% The \label command must come after the \caption command.
% \begin{figure}
%   \includegraphics{filename}
%   \caption{Some caption}
%   \label{somelabel}
% \end{figure}
```

```
%%% --------------------------------------------------------------------
%%% --------------------------------------------------------------------
%%% The number "10" that appears in the next command has no
%%% significance other than its width when printed.  It should be
%%% chosen so that it is at least as wide as any number of an item in
%%% the bibliography.  If you have 100 or more bibliography items,
%%% replace it with, e.g., \begin{thebibliography}{100}.
```

```
\begin{thebibliography}{10}


%%% The format of bibliography items is as in the following examples:
%%%
%%% \bibitem{yellowmonster}
%%% A. K. Bousfield and D. M. Kan, \emph{Homotopy Limits, Completions
%%% and Localizations}, Lecture Notes in Mathematics number 304,
%%% Springer-Verlag, New York, 1972.
%%%
%%% \bibitem{HA}
%%% D. G. Quillen, \emph{Homotopical Algebra}, Lecture Notes in
%%% Mathematics number 43, Springer-Verlag, Berlin, 1967.





\end{thebibliography}
\end{document}
```

## References


[1] American Mathematical Society, *User's guide for the `amsmath` package*, version 2.0, revised 2002-02-25. This is the file `amsldoc.pdf`, available from the AMS ftp site as `ftp://ftp.ams.org/pub/tex/doc/amsmath/amsldoc.pdf` or from CTAN (the Comprehensive TeX Archive Network) at `http://www.ctan.org/tex-archive/macros/latex/required/amslatex/math/amsldoc.pdf`.

[2] American Mathematical Society, *Instructions for preparation of papers and monographs: $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX*, November, 1966. This is the file `instr-l.dvi`, available from the AMS ftp site `e-math.ams.org`.

[3] American Mathematical Society, *Sample paper for the amsmath package*, November, 1996. This is the file `testmath.tex`, available from the AMS ftp site `e-math.ams.org`.

[4] A. K. Bousfield and D. M. Kan, *Homotopy Limits, Completions and Localizations,* Lecture Notes in Mathematics number 304, Springer-Verlag, New York, 1972.

[5] Michael Downes, *Short Math Guide for LaTeX*, American Mathematical Society, available at `ftp://ftp.ams.org/pub/tex/doc/amsmath/short-math-guide.pdf`.

[6] Leslie Lamport, *LaTeX: A Document Preparation System*, 2nd Edition, Addison-Wesley, 1994.

[7] Tobias Oetiker, Hubert Partl, Irene Hyna, and Elisabeth Schlegl, *The not so short introduction to LaTeX 2ε*, version 4.24, available from CTAN (the Comprehensive TeX Archive Network) at `http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf` and many mirrors.

[8] D. G. Quillen, *Homotopical Algebra,* Lecture Notes in Mathematics number 43, Springer-Verlag, Berlin, 1967.

[9] Kristoffer H. Rose, *XY-pic User's Guide*, version 3.7, available from CTAN (the Comprehensive TeX Archive Network) at `http://www.ctan.org/tex-archive/macros/generic/diagrams/xypic/` and many mirrors, as `xyguide.pdf`.

[10] Kristoffer H. Rose and Ross Moore, *XY-pic Reference Manual*, version 3.7, available from CTAN (the Comprehensive TeX Archive Network) at `http://www.ctan.org/tex-archive/macros/generic/diagrams/xypic/` and many mirrors, as `xyrefer.pdf`.

Department of Mathematics, Wellesley College, Wellesley, Massachusetts 02481
*E-mail address*: psh@math.mit.edu