Using the `ucsc-report` LaTeX style file for UCSC technical reports

Kevin Karplus

ABSTRACT

This technical report describes how to use LaTeX to produce technical reports in the proper format for the UCSC Baskin Center for Computer Engineering and Information Sciences series.

The style file can also be used for various other reports and articles, including theses in a format acceptable to the UCSC Division of Graduate Studies.

Two other style files are described: `ucletter` for business letters that generate University of California letterhead and `handout` for class handouts.

The report itself is written using the `ucsc-report` style file, so that the `.tex` file can be used as an example file.

The reader is expected to refer to Lamport's manual [Lamport, 1986] and to the local guide [Computer Engineering, 1990] for details on how to use LaTeX.

Contents

1. Why bother with the `ucsc-report` style file?

The question has many interpretations:
- Why write technical reports?
- Why use LATEX to produce technical reports?
- Why use this particular style file?

1.1   Why write technical reports?

Technical reports are the fastest publication medium for technical results. Fast publication protects your claim to prior discovery in a fast-breaking field, and allows you to disseminate interesting results without the usual six month to two year publication delays of journals.

Technical reports are also appropriate for documenting programs. Usually, the documentation has a small, local audience; journals and conferences are not interested in publishing detailed documentation. This technical report is itself an example of documentation that needs local distribution, but is not otherwise publishable.

Theses, particularly Master's theses, should be published as technical reports, as they are otherwise fairly difficult for others to get copies of. University Microfilms indexes and distributes Ph.D. theses, but there is a substantial delay, and the quality of their reproductions is often quite poor.

1.2   Why use LATEX?

LATEX is a macro package written by Leslie Lamport [Lamport, 1986] for Donald Knuth's TeX [Knuth, 1984]. TEX is a versatile document preparation program, but is often difficult for beginners. LATEX sacrifices some versatility to make it easier to create documents. LATEX makes chapter and section headings, tables of contents, and indices easy to include in reports. Both TEX and LATEX provide much better typesetting than competing programs, such as `troff` and SCRIBE. Changing from `troff` to LATEX can be a bit of a shock—much of the painfully acquired arcane lore is now useless. For those whose main reason for using `troff` is the the `refer` bibliography database system, a conversion program exists to change to the BIBTEX system that LATEX supports (see page 12).

The quality of the typesetting and the increased readability of the source documents easily justifies the effort of learning LATEX. The strength of TEX is particularly noticeable in setting mathematical formulae, like this one:

$$f_\nu(\bar{x}_i) = \sum_{j=0}^{\alpha_\nu} y_j^{(i)}.$$

(LATEX is also handy for producing overhead transparencies using SliTeX, but that application does not use the `ucsc-report` style file.)

LATEX provides a crude but useful language for composing diagrams. Allen Van Gelder and David Helmbold are both familiar with the `picture` environment of LATEX; I have only tried to use it once or twice. Figure 1.1 is a picture provided by David Helmbold, to give you an idea of the capabilities of the `picture` environment.

Another way to generate pictures is with `tpic`, which accepts essentially the same commands as `pic` does for generating pictures in `troff`. The `tpic` program is far more versatile than the `picture` environment, and somewhat easier to use. For more information, see the Unix manual.

It is also possible to include pictures in PostScript format generated by other programs, including `psfig` and Macintosh drawing programs. For more details, see the local user's guide [Computer Engineering, 1990], which is being prepared by Dirk Coldewey.

	LPT	RRA at least
	1	2
	2	3
	3	3
G_A	4	4
	5	4.4
G_B	6	2
	7	1
	8	2
	9	1
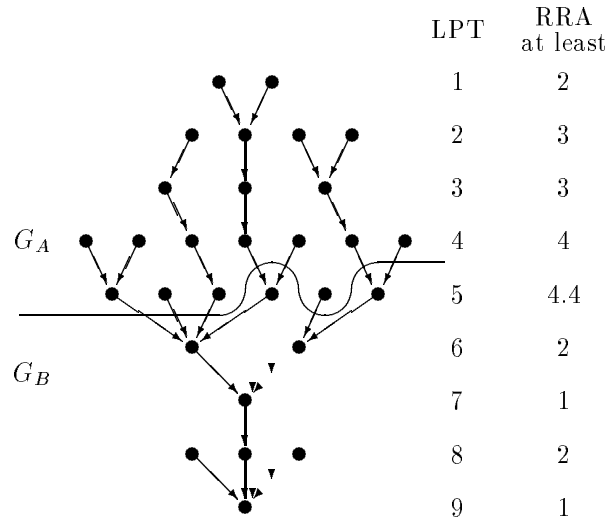
Figure 1.1: A diagram created by David Helmbold using LaTeX's `picture` environment.

1.3   Why use the `ucsc-report` style file?

LaTeX comes with several style files for different sorts of documents, but none of the supplied style files is quite right for UCSC technical reports. The `ucsc-report` style file was modified from the standard `report` style file to meet the needs of our technical report series. With the `thesis` option, the `ucsc-report` style file is ideal for typesetting theses at UCSC.

The new `article` and `ieee` options make it good for typesetting conference and journal articles– at least as good as Lamport's `article` style.

Some of the notable differences between the `ucsc-report` style file and Lamport's `report` style file are:

- The default format has been changed to 11-point type size, two-sided printing, and headings at the top of each page. Unlike other style files, the first page of a chapter has a normal heading at the top, rather than centering the page number at the bottom. Having page numbers in different places on different types of pages seems strange to me, but is standard practice for books. I chose to keep the page numbers in the upper, outer corners, even for chapter beginnings. If you prefer the conventional style, add `\thispagestyle{plain}` after each `\chapter` heading.

- The margins have been reduced to fit more text on a page. All margins are now one inch, except the gutter margin, which is $1\frac{1}{2}$ inches, to allow for the binding. In the default two-sided format, odd-numbered pages are right-hand pages with a wide left margin, and even-numbered are left-hand pages, with a wide right margin. For one-sided printing, the left-hand margin is always the wide one.

- Title pages now fit the UCSC covers. Both the left and right margins are $1\frac{1}{2}$ inches, so that the title, author, and date will appear properly centered in the window of the cover.

  The back of the title page should be left blank in two-sided printing, because the first page after the title is a right-hand page. If you want to have an errata or dedication page on the back of the title page, put it after the title page, and make it have page number 0. After the `\maketitle` that creates the title page, you could put `\setcounter{page}{0}` `\thispagestyle{empty}` to have an un-numbered left-hand page.

- Chapter and section headings are more compact than in the `report` style. As before, chapters start a new page, but not necessarily a right-hand page. If you do not want chapters as your main division, you can specify the `nochapter` option in your `\documentstyle` command, to make sections the highest level of organization.

- Draft copies are labeled "Draft Copy—Do not circulate." and dated on each page.

- The table of contents and theorem environments are punctuated better than in the `report` style file. The theorem numbers now have colons after them, and chapter numbers in the table of contents have periods.
- Normal indentation is used for the first paragraph of each section.
- The penalties for club lines, widow lines, and page breaks after hyphens have been increased to make bad page breaks less likely.
- The tolerance for bad line breaks has been increased slightly to reduce the number of warning messages about bad line breaks (that is, fewer `Overfull \hbox` error messages from TEX). The tolerances have been increased even more in the bibliography section, as it is usually very difficult to find good line breaks in bibliographies.
- Hyphenation is tried even for words beginning with upper-case letters. If you get a bad hyphenation, you can use TEX's `\hyphenation` command to give the correct hyphenation. If you get many bad hyphenations of capitalized words, you can turn off hyphenation for capitalized words with `\uchyph=-1`.
- The glue between paragraphs is stretchier than in Lamport's `report` style, so that better page breaks can be found by TEX's page-breaking routines.
- Separation between items of a list and around lists has been reduced.
- The definitions of `\{` and `\}` , which print the braces { and } have been changed to produce { and } when using the typewriter font.
- Indexing commands have been incorporated into the style file and an auxiliary program provided to make indices automatically.

2. How can I use the `ucsc-report` style file?

2.1 Look at the source file for this technical report

The easiest way to use any document formatter is to modify an existing input file that is known to work correctly. By treating most of the commands as ritual magic, you need to learn only those commands that are essential for the material you are presenting. One such example is the source file for this technical report (`ucsc-tr.tex` and associated bibliography `ucsc-tr.bib`), available on most of the UNIX machines in the Baskin Center, usually in `/usr/local/TeX/lib/inputs`.

You should get the tech report from Gerri McLellan, not print your own on the laser printer, as the duplicated copies are far cheaper than the laser printer copies. However, if the report goes out of print for a while, you can generate your own copy with the following sequence of commands:

```
cp /usr/local/TeX/lib/inputs/ucsc-tr.* .
latex ucsc-tr
bibtex ucsc-tr
latex ucsc-tr
makeindex ucsc-tr
latex ucsc-tr
latex ucsc-tr
dvips -Paca ucsc-tr.dvi
```

The first call to `latex` generates the auxiliary file and the index file, then `bibtex` generates the bibliography. The next call to `latex` includes the table of contents and the bibliography in the report. Then `makeindex` generates the index and fixes citations. The next call to `latex` includes the index and the final one fixes up the table of contents and cross-references. The call to `dvips` prints the file on the laser printer named `aca`—you may choose any output device that understands PostScript commands. Don't forget to delete all your copies of the `ucsc-tr` file when you are done.

The following files are derived from the `ucsc-tr.tex` file and the `ucsc-tr.bib` bibliography file:

- `ucsc-tr.aux` contains citations, references, labels, and other auxiliary information made by LaTeX.
- `ucsc-tr.toc` is the table of contents file, made by LaTeX.
- `ucsc-tr.idx` contains index entries made by LaTeX.
- `ucsc-index.tex` contains the index generated from `ucsc-tr.idx` by `makeindex`.
- `ucsc-tr.bbl` contains the processed bibliography, made by BibTeX.
- `ucsc-tr.dvi` is the final output file, ready for printing with `dvips`. Because `.dvi` files are large and easily re-created, they should be deleted after printing.

2.2 The parts of a LaTeX file

A LaTeX file begins by requesting a document style. To request the `ucsc-report` style, begin with `\documentstyle{ucsc-report}`.

The `\documentstyle` command can take several optional arguments. If you specify more than one option, separate the options with commas. The options for `ucsc-report` include:

draft If you are not finished with the report, you may prefer to use `\documentstyle[draft]{ucsc-report}`, which will print a notice "Draft Copy—Do not circulate." on the bottom of each page. The `draft` option also marks lines for which TeX reported an `Overfull \hbox` by putting a black box in the right-hand margin. Note that the right-hand margin is used, not the outside margin. Because draft copies are rarely bound, it shouldn't matter that the right-hand margin is the gutter margin on even-numbered pages.

oneside If your report is not going to be reproduced double sided, you can request that the wider gutter margin always be on the left with the command `\documentstyle[oneside]` `{ucsc-report}`. Page numbers will always be in the upper right corner.

twoside For two-sided reproduction, the even-numbered pages have the gutter margin on the right, and the page numbers are put in the upper left corner. The `twoside` option is the default if neither `oneside` nor `twoside` is specified.

9pt, 10pt, 11pt, 12pt The default type size for the main text is eleven points, but you can specify twelve-point, ten-point, and nine-point fonts in the `\documentstyle` command. You should use twelve-point fonts if you plan to photo-reduce the document. Ten-point fonts are useful if you need to cram a lot of information into a specific page limitation. If you use ten-point fonts, you should consider using a two-column layout, as the standard six-inch long lines get a little hard to read in small fonts. For nine-point fonts, two-column setting is essential.

twocolumn The `twocolumn` option requests that the main text be set in two columns, each slightly less than three inches wide, rather than in a single six-inch wide column. The title page is not affected by this option, and the index is always set in two columns. Because setting in narrow columns is more difficult, the `twocolumn` option changes TEX's tolerances to accept somewhat looser and tighter lines, but you should probably reduce the typesize to ten points (with the `10pt` option) to cut down further on the number of overfull hboxes.

onecolumn The `onecolumn` option specifies that the main text is to be set in a single column. It is the default if neither `onecolumn` nor `twocolumn` is specified.

nochapter The main division of a technical report is usually a chapter, but for short reports, it may be desirable to make the highest-level division a section. Specifying `nochapter` in the `\documentstyle` command corrects the section, figure, and table numbering, and makes other minor corrections needed when there are no `\chapter` commands.

The main advantage of using `nochapter` is that chapters always start on a new page, but sections do not have to. If your main divisions are short, calling them chapters can leave you with a lot of white space.

chapter The `chapter` option makes chapters be the highest level division of a report. It is the default if neither `chapter` nor `nochapter` is specified.

article The `article` option produces a document with the text starting on the same page as the title and author, with the title information centered at the top of the page. The `article` option implies the `nochapter` option.

ieee The `ieee` option tries to match the specifications given by the IEEE Computer Society Press (for articles in the EuroASIC-92 Conference). It implies `twocolumn`, `article`, `nochapter`, `oneside`, `seqcaption`, and `10pt`. It also changes the margins, changes the font used for captions, suppresses headers and page numbers, and makes several other minor changes. It may be the best starting point for generating camera-ready copy for conference proceedings.

thesis The `thesis` option, for typesetting Master's and Ph.D. theses at UCSC, is described in detail in Section 2.7.

seqcaption The `seqcaption` option causes figures and tables to be numbered sequentially from the beginning of the document, rather than using a two-part number that includes the section or chapter number.

index If you want to generate an index with the macros provided, you must include the `index` option in the `\documentstyle` command. Further explanation of the index macros is given in Section 2.6.

noindex If you do not want an index, you can leave out the `index` option or specify `noindex`. You may still use the various indexing macros, but they will have no effect.

showindex The `showindex` option is an experimental option to put a notation in the margin for each index entry. If you specify `showindex`, no index will be built, even if you specify `index` as well. Unfortunately, the current version of `showindex` is built on top of the built-in `\marginpar` command, which is not really capable of handling the task. Until a better implementation of `showindex` is written, it is probably not worth bothering with.

The problem with \marginpar is that it cannot properly handle multiple marginal notes that are too close together. Because index terms often cluster, the density of \marginpar entries gets too high, and the notes end up in apparently random places.

Everything from the \documentstyle line until the line containing \begin{document} is the *preamble*. The preamble usually contains global definitions that affect everything in the document.

After the preamble comes the title page, followed by a request for the table of contents, the chapters of the report, a request for the bibliography, and a request for the index. A LaTeX file ends with the \end{document} command.

2.3 The title page

The ucsc-report style file has made some substantial changes in the way title pages are produced. The following commands are provided to accept the information needed to build the title page:

- \title
- \author
- \reportnumber and \noreportnumber
- \address
- \abstract
- \keywords

For single-author papers, simply type the appropriate information in braces after each command and follow the commands with \maketitle. If a title is too long to fit on one line in the window, the command package will automatically break it, but not in a semantically reasonable place. You can specify your own line breaks in a title by using the \\ command to force a line break.

For example, this paper begins with

```
\title{Using the {\tt ucsc-report}\\
\LaTeX{} style file\\
for UCSC technical reports}

\author{Kevin Karplus}

\reportnumber{UCSC-CRL-92-47\\
supersedes UCSC-CRL-90-25 and UCSC-CRL-87-10}

\abstract{...
}
\keywords{\LaTeX, style file, typesetting, document compiler}
\maketitle
```

No address is given, which means that the address defaults to the address for the Baskin Center: \address{\baskin}. Besides the Baskin Center address, you can use the \ce command for Computer Engineering or \cis for Computer and Information Sciences as an argument to \address—or you can provide a complete address as an argument.

The report number is assigned by Gerri McLellan—you should request one after you have finished the text, so that she is not left with holes in the sequence if you don't finish as soon as you expect. If you are using the ucsc-report style for a document which is not a tech report, you can omit the report number by specifying \noreportnumber some time before the \maketitle.

A special footnote command \thanks is provided for use on the title page. Lamport suggests its use "for an acknowledgement of support, an author's address, etc.", but these different pieces of information belong in different places. In the ucsc-report style, use \thanks for acknowledgements of support only. For example, had I wished to thank someone for supporting this report, I would have changed the \author entry to \author{Kevin Karplus\thanks{Partially supported by a grant from ... .}}

Some people like to put a job title at the beginning of the address. If you want to, you may either use the `\address` command (`\address{Assistant Professor\\ \ce}`) or the special `\jobtitle` command (`\jobtitle{Assistant Professor}`). The `\jobtitle` command is no longer recommended, and is included only for compatibility with older versions of `ucsc-report`.

For a paper with multiple authors, you have to do a bit more work to associate the correct titles and addresses with the individual authors. If all authors have the same address, you can use LaTeX's standard way of specifying multiple authors: `\author {Author One \and Author Two}`. Lamport suggests using `\thanks` for giving addresses for different authors, but the `\thanks` command will not put the information in the right place on the title page, so you have to make your own footnote marks. I recommend something like this:

```
\author{Author One$^\ast$
\and
Author Two$^\dag$}
\address{$^\ast$ \ce \\ \ \\
$^\dag$ \cis}
```

Novices may need some explanations to understand that mass of LaTeX commands. A complete explanation would take too much space, but here are a few hints:

- The command `\\` requests a new line. However, you can't put two `\\` in a row, which is why the example uses `\\ \ \\` to provide a blank line. The extra forced space separates the two `\\`.

- The command `$^\dag$` enters math mode, puts a dagger as a superscript, and leaves math mode again.

- The `\address` information is preceded by the appropriate footnote markers.

If you have to mention different grants for different people with different addresses, you should use the `\thanks` command to generate the footnotes acknowledging the grants, and put the same footnote markers in front of the appropriate addresses. The `\thanks` command uses symbols for footnote marks, in the order given in the LaTeX manual [Lamport, 1986, page 175]:

1. ∗ created with `\ast`.

2. † created with `\dag`.

3. ‡ created with `\ddag`.

4. § created with `\S`.

5. ¶ created with `\P`.

6. ∥ created with `\|`.

7. ∗∗ created with `\ast\ast`.

8. †† created with `\dag\dag`.

9. ‡‡ created with `\ddag\ddag`.

Every technical report should have an abstract. In this style file, the abstract is passed as the argument to the `\abstract` command, not using the `abstract` environment as described in the LaTeX manual [Lamport, 1986, page 84]. If you want your abstract to be called something other than *abstract*, you can use an optional argument to give it a different name. For example, you can now say `\abstract[Executive Summary]{...}` to call the abstract an executive summary.

The abstract is put in the tech report catalog, to help people order the most interesting reports, so it helps to be as complete and informative as possible. Your abstract can have as many paragraphs as you like, but should be short enough to fit on the title page. The first paragraph will not be indented, but subsequent ones will be.

You will help librarians enormously if you add a `\keywords` field to your title page. It isn't necessary to repeat words from your title, but it is helpful to include any phrases from the abstract or text that would help in finding the report using a keyword search.

2.4 The table of contents

The table of contents is produced by the usual LaTeX mechanism, directly from the chapter and section headers. To get the table printed in the right place in your report, include the command `\tableofcontents` after `\maketitle` but before the first `\chapter` command. Because changes to the table of contents may change its length, changing the numbers of later pages, you may need to run LaTeX twice after the last changes to the text to be sure the table of contents is correct. LaTeX warns you if it thinks that cross-references may have changed—if you don't get the warning, it should be ok.

2.5 Chapters and sections

The main divisions of a technical report are normally chapters started with the `\chapter` command, but you can use larger groupings called "parts" by using the `\part` command. Numbered sections can be made inside chapters with the `\section` command.

2.5.1 Also subsections

Subsections, such as this one, can be started with `\subsection`.

And (unnumbered) subsubsections

You can even have subsubsections, such as this one, started with `\subsubsection`. If you want your subsubsections to be numbered, add the command `\setcounter{tocdepth}{3}` to the preamble of your document.

You can't go any deeper than subsubsections using the `ucsc-report` style file, unless you want to number paragraphs with the `\paragraph` command. If you feel the urge to have (sub)^{n}sections, you'll have to define your own commands for them. Better yet, consult a technical writing consultant, to see if you can have the urge removed.

2.6 Making an index

Indices are useful things, but can be difficult to write. Not every technical report needs an index, but reports that lack a needed index are more common than reports with a superfluous one. User documentation for a program almost always needs an index.

You could use LaTeX's built-in index capabilities [Lamport, 1986, pp. 78–79, 153, 186, 189], but you'd have to write your own program to sort the entries, merge adjacent page numbers, choose the correct fonts, and format them correctly. I have written some commands and scripts to generate indices automatically. My commands are a bit idiosyncratic, but were intentionally made easy to modify. You can use them as-is, or modify them to suit your own tastes. To use my index commands, include the option `index` in the `\documentstyle` command, as described on page 6.

The index commands do not print anything at the place in the report where they occur. Instead they send information to a *<file>*`.idx` file, which is later processed by `makeindex` to produce the index. The `csh` script `makeindex` should be in the same directory as `tex` and `latex`—usually `/usr/local/bin`. The index commands provided are

`\indexone{key}` puts the key into the index with a reference to the current page. The first letter of the key will be capitalized automatically. This may be the only index command you need.

`\indexonenopage{key}` puts the key into the index with no reference to the current page. This is useful for adding simple cross-references to the index.

`\indextwo{key}{print-as}` puts the `print-as` field into the index, sorted as if it were `key`. This
is often used for entering words that begin with special characters. For example, the index
entry for `\indextwo` is `\indextwo{indextwo}{\verb+\indextwo+}%`. Note the use of LaTeX's
verbatim command `\verb`, with matching plus signs to delimit the string to be quoted. As
another example, `\indextwo{foo}{foo}` will make sure that "foo" is put in the index without
being capitalized.

`\indextwonopage{key}{print-as}` does the same job as `\indextwo`, but does not include the page
number. Its main use is for providing cross references that need formatting. It can also be
used for adding formatting information for main keys that do not need page numbers—see
`\indexsub` below.

`\indextt{key}` is a special case of `\indextwo` for programmers. The key is sorted normally, but
printed in the typewriter font `\tt`.

`\indexsub{key}{subkey}` provides for a key and subkey. Both the key and the indented subkey
will appear in the index, with a page entry made for the subkey, but not for the key. If the
key should also be indexed at this point, use `\indexone{key}` as well.

If the subkey needs to be formatted, you have to use `\indextwo` instead. For example, see
the *address commands* entry in the index to this report. The entry for `\ce` was created with
`\indextwo{address commands*ce}{\verb+\ce+}`, using the * to separate the key and the
subkey in the first field.

If the main key needs formatting, add an `\indextwonopage` command for it somewhere in the
file.

I have found it easiest to put each indexing command on a separate line, and to end the line
with TeX's comment character "%". This prevents extra spaces from appearing in the output, and
makes the `.tex` file easier to read. Warning: don't end a paragraph with an indexing command,
because some bug in either the index commands or LaTeX occasionally causes an extra blank space
to be added at the end of the paragraph. If the text of the paragraph just fills the last line, the
blank space becomes a full blank line.

Because the sorting and index-building programs use "*" as a delimiter to separate a key from
a subkey, and "!" to separate other fields, indexing text containing either character requires some
trickery. Perhaps the easiest way is to add two lines in your preamble:

`\def\aster{*}`
`\def\bang{!}`

and use the commands `\aster` and `\bang` in your index entries where you want the special characters
to be printed. Another solution is to re-write the index commands, the `makeindex` script, and the
`texindex.awk` program to use different characters for the delimiters. In the `texindex.awk` file, the
constant `FS` would need to be changed, and in `makeindex` both the "*" as the first argument of `tr`
and the `-t'\!'` as the first argument of `sort` would need changing.

To generate an index from *<file>*`.tex`, you need to run `latex` *<file>* to generate a *<file>*`.idx` file,
then run `makeindex` *<file>* to generate *<file>*`-index.tex`.

At the end of the *<file>*`.tex` file, include the command `\input{<file>-index}` to read the index
file created by `makeindex`. Once the index has been created, you need to run LaTeX again to include
the index in the report. If the index is properly placed at the end of the report, changes to the
index will not change the page numbers of anything else in the file. Be sure that you run (or re-run)
`makeindex` after any changes to the file. If you have a table of contents, you should be sure that
LaTeX has been run with the full table of contents before running `makeindex`. If you are paranoid
about all the various things that could move page numbers, you might want to run the following
sequence of commands before your final printing (assuming you are working on a file called `foo.tex`):

```
latex foo
bibtex foo
latex foo
latex foo
```

```
makeindex foo
latex foo
```

The first time you run LaTeX (before running `makeindex`) you will not have an index file *<file>*-`index.tex` file. When LaTeX complains and asks for another file name, type a control-D. Once the *<file>*-`index.tex` has been created by `makeindex`, LaTeX will not complain any more.

2.7 Printing UCSC theses

Theses at UCSC, as at most universities, have a number of strange formatting requirements—some justified, others perhaps not. To make it easier for students to print their theses, a `thesis` option has been added to `ucsc-report`. Because I believe that most theses should also be published as tech reports, I have made `thesis` be an option in `ucsc-report` rather than creating is as a separate style file—this should make it easy to prepare a document to be published in both ways.

A dummy thesis has been checked by the Division of Graduate Studies, who check thesis formatting. They saw no problems with the formatting of the dummy, but you are ultimately responsible for your thesis and should read their guidelines carefully.

Using `thesis` as an argument to `\documentstyle` makes the following changes:

- The `oneside` option is forced.
- The `chapter` option is forced.
- The `\baselinestretch` is set to 1.6, to simulate double spacing, except for the bibliography, which remains single spaced inside each entry. No attempt is made to reset the `\baselinestretch` back to 1.0 in other places where double spacing is not required (captions, tables, indented quotations, footnotes, and so forth), but you can do this yourself by including the command `\nostretch` inside an environment or pair of braces delimiting the paragraph to be set normally. The `\nostretch` macro is useful even if you aren't using the `thesis` option, if you have changed `\baselinestretch` yourself.

  Captions and footnotes will be single-spaced if the `\nostretch` macro is included inside them. Note that entire paragraphs are set with whatever `\baselinestretch` is in effect at the end of the paragraph, and so asserting `\nostretch` for just the middle of a paragraph will have no effect.

- Top and bottom margins are increased to 1.25 inches to meet the standards. The standard left and right margins are already ok.
- The `\maketitle` macro is changed to create a title page in the required style. There are several changes to related macros:
  - The `\date` macro is ignored. Instead you should use of one of the four macros `\fallquarter`, `\winterquarter`, `\springquarter`, or `\summerquarter` before issuing the `\maketitle` command, to get the correct month and year on the title page. You can specify the year to be different from the current year with an optional argument to the appropriate quarter macro: `\fallquarter[1999]`.
  - One of the macros `\PhD`, `\MS`, or `\MA` should be called with a required argument giving the field of the degree. For example, a computer engineering student might specify `\MS{Computer Engineering}`
  - The `\thanks` macro is not supported—use a separate acknowledgements section.
  - The address is not needed, and so `\address` is not used.
  - You need to specify your advisors with the `\advisors` macro, which takes any number of arguments: `\advisors{Real Advisor}{Another Member}{Last Member}`
- A new `\copyrightpage` macro is available for creating the copyright page. It should be called immediately after `\maketitle`. If you are not copyrighting your work, then leaving out the `\copyrightpage` macro will increment the page numbers correctly, but the blank page is not printed—don't forget to insert a blank piece of paper yourself!

```
\maketitle
\copyrightpage % copyright is optional
\tableofcontents
\listoffigures  % optional
\printabstract

\chapter*{Acknowledgements}
\addcontentsline{toc}{chapter}{Acknowledgements}
I want to thank whoever deserves it.

\clearpage\pagenumbering{arabic}
```

Figure 2.1: A fragment of a thesis, showing how to order the commands for creating the front matter using the `thesis` option.

- The page numbers are still in roman numerals after `\maketitle`, so that you can include lists of figures, lists of tables, your abstract, a dedication, or acknowledgements. The title page, table of contents, and abstract are required, but the rest of the front matter is optional. The front matter should come in the order: title page, optional copyright page, table of contents, optional list of figures, optional list of tables, abstract, optional dedication, and optional acknowledgements. Although the lists of figures and tables are probably optional (the guidelines are a bit unclear on this point), they are highly recommended, and take no extra effort on your part.
- The abstract is no longer on the title page. An abstract is required for theses, and is limited by the Graduate Studies Division to 350 words. The style file does not enforce this limitation, so you must do your own word count. Use the `\abstract` command as usual, but put a `\printabstract` command where the abstract belongs. The `\printabstract` command takes care of turning off the printing of the page numbers for the abstract, as required by the Division of Graduate Studies.
- To switch to arabic numerals for page numbers in the main body of the thesis, use the following sequence of commands: `\clearpage\pagenumbering{arabic}`.

A dummy thesis `dummy-thesis.tex` has been created for you to modify—it should be in `/usr/local/lib/tex/inputs`. Figure 2.1 shows a typical sequence of commands for making the front matter, taken from that example.

2.8   Bibliographies

The best way to create a bibliography is to use Oren Patashnik's BIBTEX [Lamport, 1986, Appendix B]. To use BIBTEX, you need to create a `.bib` file containing all the papers you wish to cite. You may include as many other papers as you like in each `.bib` file, and you may use multiple `.bib` files. Research groups often share a common `.bib` file.

On a UNIX system, to use a `.bib` file that isn't in the same directory as the `.tex` file, you need to set the `BIBINPUTS` environment variable. For example, in my `.cshrc` file, I use the command

```
setenv BIBINPUTS .:/a/faculty/karplus/tex
```
to make BIBTEX search my `tex` directory for my standard bibliography files.

BIBTEX provides roughly the same capabilities as the `refer` bibliography database for `troff`, but in an incompatible format. Converting small databases is easily done with a text editor, but for those who are avoiding switching from `troff` mainly because of `refer`, a conversion program (`r2bib`) exists to change to the BIBTEX system. Because I have no `refer` databases, I have not used the `r2bib` conversion program, and it may not be installed on all systems. David Helmbold has experimented with an `awk` program to convert bibliography databases from `refer` to BIBTEX format. Ask him for advice before undertaking a massive conversion project.

I have found it useful to gather all my bibliographic references into only one or two `.bib` files, which I keep in the directory `~/tex`. That way I can reuse references that I have found useful, without having to search through my entire set of directories for the paper in which I cited them. Of course, if you are working on a joint paper with another author, it may be more convenient to extract just those references you actually use, so that the `.bib` file isn't any bigger than it needs to be.

Instructions for using BibTeX are included in the LaTeX manual [Lamport, 1986] and will not be duplicated here. However, there are several new bibliography style files that are not included in the manual. Look for files ending with `.bst` in the directory which BibTeX searches (probably `/usr/local/lib/tex/inputs`), and read the comment at the beginning of each to see when it is appropriate.

I recommend using `\bibliographystyle{aaai-named}` for the citation style, as it creates the most readable of the currently available citation formats. This long style is particularly appropriate for theses, as they are often read on microfiche or microfilm, which makes turning to the bibliography rather tedious. The similar bibliography style `named` requires that you include `named` as an option in the `\documentstyle` command. The `named` option is a true substyle (that is, there is a separate `named.sty` style file).

The `aaai-named` bibliography style requires using version 0.99 or newer of BibTeX.

If you are short on space, then `\bibliographystyle{alpha}` provides a more condensed format. Some authorities recommend using an alphabetically ordered bibliography and simple numerical citations, as there is less temptation then to treat the citations as part of the sentence. For numeric citations, the `plain` style can be used.

The `\bibliography` and `\bibliographystyle` commands should appear in the `.tex` file where you want the bibliography to appear. Normally, this will be after the last chapter, but before any appendices.

2.9 Printing the final output

The `.dvi` file created by LaTeX is device-independent, and has to be converted before being output to a printer. Our laser printers expect PostScript commands, and so the program `dvips` should be run to do the conversion. You can use any of the PostScript printers, but some of the older ones do not have enough memory for TeX files with many font changes, and will not print the entire file.

The `man` pages for `lpr` claims that the conversion is done automatically if you use the `-d` switch of `lpr`, but this forces the conversion to be done on whatever computer the printer spool queue is kept for that printer. Forgetting the `-d` switch can tie up the printer for hours, wasting paper for no useful purpose.

You will get much more reliable results if you invoke `dvips` directly. To print the output from `latex foo.tex`, you give the command

    dvips -Paca foo.dvi

The `.dvi` files themselves can be quite large, and are easily re-created, and so should be deleted after printing. The log files (`.log` and `.blg`) should also be deleted after you have examined them.

2.10 Reporting bugs

If you have difficulties using `ucsc-report`, find a bug in the style file, or have suggestions for improvement, please talk to me or send electronic mail to `karplus@ce.ucsc.edu`.

If you can't find `ucsc-report.sty` and `ucsc-report.doc` (the commented version of it), contact Dirk Coldewey (`dirk@cse`). Also contact Dirk if all you can find is the old (1987 or 1990) version. Problems with the installation of LaTeX or with style files other than the ones in this report should also be directed to Dirk.

3. Other style files

3.1 Writing letters with `ucletter`

The `ucletter` style file was modified from the standard `letter` style [Lamport, 1986, Section 3.8]
by several people, including Pak Chan, Allen Van Gelder, and me. The `ucletter` style is especially
designed to create University of California letterhead, complete with seal. It should only be used
for UC business—for other purposes, try using the standard `letter` style.

The `ucletter` style has options for each of the main UC campuses (but not the medical centers).
Only the `ucsc` option has been used extensively—the addresses and zip codes of the other campuses
should be checked carefully before using them. You must specify one of the campuses as an option
to the `\documentstyle` command, or the letterhead will not be generated correctly.

In addition to the `ucsc` option, you may specify one of `ce`, `cis`, or `baskin` as a `\documentstyle`
option, to customize the letterhead for the particular department.

To create a business letter with UC logo heading, you need to provide information in your
preamble with the following macros:

- `\location` takes one argument—the room number. The location information is optional.
- `\email` takes one argument—your electronic mail address. The electronic mail address is
  optional, but provides a good way for people to reach you.
- `\telephone` takes one argument—your phone number. You should give the complete phone
  number, not just your extension, unless the letter is being sent only to someone else on campus.
- `\name` and `\signature` each takes a single argument—your name. You may provide either one
  or both. The name you provide to `\signature` is used to generate the closing, unless it isn't
  provided, in which case the name provided in `\name` is used.
- `\dept` takes a single argument—the name of your unit. If you use the `ce`, `cis`, or `baskin`
  option, then you do not need to have a `\dept` command in your preamble.

I have found it useful to create a file called `~/tex/return-addr.tex` that contains the information
I need for almost every letter. Because I have set the path in the `TEXINPUTS` environment variable to
include my `tex` directory, all I need to put in the preamble for letters is `\input{return-addr}`. For
more information on setting up the `TEXINPUTS` environment variable, see the local guide [Computer
Engineering, 1990]. Figure 3.1 shows my return address file.

The new `ucletter` style file has two main environments: `letter` and `narrowletter`. The two
environments are used identically—the only difference is in the size of the side margins. For very
short letters, the `narrowletter` environment is probably preferable, as the `letter` environment is
set up primarily for letters of a page or more.

A letter is started with either `\begin{letter}` or `\begin{narrowletter}` after the initial
`\begin{document}`. You may have multiple letters in the same file, by having multiple `letter`
or `narrowletter` environments. The `\begin{letter}` should be followed by the name and address
of the recipient of the letter, in braces. This will be used as the inside address of the letter, and will
be used for the address label if `\makelabels` is included in the preamble. Warning: `\makelabels`
has not been tested—it is not known whether the layout corresponds to any known peel-off label
size.

The first thing inside the letter should be an `\opening` command, with a single argument giving
the complete greeting line of the letter, including any ending punctuation. After the opening, you
provide the main body of the text, followed by `\closing`, which takes a single argument giving the
closing line.

After the closing, you can include annotation commands:

- `\cc` takes one argument, the list of people who are to be sent copies of the letter.
- `\encl` and `\encls` each take one argument, the name of an enclosure or list of enclosures.
- `\annotation` takes two arguments, the first is the type of annotation, the second is text of the
  annotation. All the preceding annotations are defined by using `\annotation`.

```
\name{Kevin Karplus}
\email{Internet: {\tt karplus@ce.ucsc.edu}}
\location{Applied Sciences, Room 315b}
\telephone{(408) 459-4250}
```

Figure 3.1: Sample return address file for use with the `ucletter` style file.

```
\documentstyle[ce,ucsc]{ucletter}
\input{return-addr}

\begin{document}

\begin{narrowletter}{Mr.~Fat Cat\\
Big Business\\
Easy Street\\
Somewhere, CA\ \ 66666
}

\opening{Dear Fats,}

Send money!  Thanks.

\closing{Sincerely,}

\end{narrowletter}
\end{document}
```

Figure 3.2: A short letter created using the `ucletter` style file. Note the use of a `return-addr.tex` file, as in Figure 3.1. Note the use of two forced spaces before the zip code, to move it out a bit from the state name.

- \ps takes no arguments—everything that follows is some sort of postscript. You have to add whatever formatting you want it to have. This command does not seem particularly useful to me—you can always go back and insert your afterthoughts in the right place in the letter.

An example of a complete, short letter is given in Figure 3.2. It assumes you have set up a `return-addr.tex` file as shown in Figure 3.1.

3.2   Class handouts with `handout`

3.2.1   Purpose of the style file

The `handout` style file was created for the handouts in CE 185, the technical writing class. Dan Scripture and I wanted the handouts to look particularly good, to encourage students to be pay attention to how their own writing looked.

The style file is intended for handouts that will be printed double-sided. The left-hand margin of odd-numbered pages and the right-hand margin of even pages (the *gutter* margin) is made wider to allow students to punch the handout and put it in a binder.

The style file may be used as is, but I am not willing to modify it to suit the different tastes of all instructors. If you want a different handout style, copy the style file and modify it yourself. Bug reports are welcome, and bug fixes even more welcome.

If you can't find the `handout` style file and the commented version of it (`handout.doc`) in the system directory `/usr/local/TeX/lib/inputs`, contact Dirk Coldewey (`dirk@cse`).

3.2.2   Key points

The style file should be installed as `handout.sty` on all the Baskin Center machines that run LaTeX.

The style file is similar to the `article` style described in the LaTeX manual. The main differences are that there is no special handling for titles and authors, and that special commands are used to create the header and footer lines.

The special commands each take a single argument, which is used to fill in one of the fields in the header or footer format.

`\instructor` the name of the instructor.

`\course` the department and number for the course.

`\quarter` the quarter and year of the course.

`\title` an abbreviated title to put in the header of each page.

`\displaytitle` a long form of the title to display centered in a large font at the top of the first page. If not given, the +title+ value is used. The display is produced by `\maketitle`, and can be suppressed by omitting the `\maketitle`.

`\hotype` the type of handout. I generally use `Assignment`, `Info`, and `Exam`, but you can use whatever classification scheme you prefer—including leaving this field blank.

`\honumber` the number within the classification for this handout.

`\due` the date (and time) when assignments are due. If there is no due date, you can omit the `\due` command.

Acknowledgements

References

[Computer Engineering, 1990] Computer Engineering. Using LaTeX at the UCSC Baskin Center. Technical Report UCSC-CRL-90-??, Board of Studies in Computer Engineering, University of California at Santa Cruz, Santa Cruz, CA 95064, sometime 1990. Until the local guide is released as a tech report, draft copies will be available from Dirk Coldewey.

[Karplus, 1987] Kevin Karplus. Using the ucsc-report LaTeX style file for UCSC technical reports. Technical Report UCSC-CRL-87-10, Board of Studies in Computer Engineering, University of California at Santa Cruz, Santa Cruz, CA 95064, August 1987.

[Knuth, 1984] Donald E. Knuth. *The TeXbook*. Addison-Wesley Publishing Company, Reading, MA, 1984.

[Lamport, 1986] Leslie Lamport. *LaTeX, a Document Preparation System*. Addison-Wesley Publishing Company, Reading, MA, 1986.

Appendix A. Brief sample appendix

Appendices are usually used for tables, program code, and other interesting data that is not included in the main body.

This one is included to make the source file show how to generate an appendix, and to remind the readers that appendices come after the bibliography, but before the index.

To start labeling chapters as appendices, simply include the command \appendix before the \chapter that starts the first appendix. If you are using the nochapter option, the appendices are sections, rather than chapters.

Appendix B. Changes from 1990 version of `ucsc-report`

The `article` and `ieee` options have been added, to make it easier to turn technical reports into conference and journal papers.

The new `seqcaption` option to `\documentstyle` makes the numbering of figures and tables be sequential, rather than starting over with each chapter (or section, if the `nochapter` option was selected).

Several spacing parameters have been changed to make technical reports denser, thus using up less paper. In particular, the spaces in bibliographies are narrower, the item separation parameters for list elements are much smaller, and a new `9pt` option has been added, for people who are really cramming things in. The `9pt` option should not be used unless the `twocolumn` option is also used. The tightening of the typesetting is very useful for conference papers with very small page limits.

Citations with long names now allow line breaks in the middle of the name.

The `\TODO` macro has been fixed to put the marginal stars in the right place.

The index macros have been modified slightly to compensate for changes in the underlying LaTeX code.

The `\advisors` macro in the `thesis` option now takes any number of advisors, and the title page for Master's theses now says "thesis" rather than "dissertation".

The handout style now accepts a `\maketitle` command to display a title at the top of the first page.

Appendix C. Changes from 1987 version of `ucsc-report`

The version of the `ucsc-report` style file is different in several ways from the version described in the previous report [Karplus, 1987]. This appendix describes the more noticeable of those differences. It also lists some minor new features that were not included in the main body of the text.

The new version of `ucsc-report` is a single `.sty` file, with all options included in the file, rather than using substyles. This makes it much easier to send the style file to a collaborator at a different institution.

The `thesis` option has been added to make typesetting UCSC theses in LaTeX easy.

A new option, `nochapter`, makes sections the highest division, rather than chapters. The bibliography, index, and appendices become unnumbered sections, rather than unnumbered chapters.

The page numbers of the first pages of chapters are now displayed in the standard position at the outside corner at the top of the page, rather than being centered at the bottom of the page, as is conventional for books. This practice differs from other LaTeX style files, which follow the conventional form.

The bibliography now has the label `references`, so that it can be referred to with the `\pageref` macro. For example, the bibliography for this report is on page 18.

The top and bottom margins are a quarter inch smaller than before, to reduce the number of pages in a report, thus reducing the printing and mailing cost.

Minor changes were made to the layout and specification of the title page:

- The default address on the title page was changed to `\baskin`.
- The `\jobtitle` field was made optional (default, not used).
- A new macro, `\noreportnumber`, was added to suppress the report number field for publications that are not tech reports.
- The `\abstract` macro was given an optional argument, to allow the abstract to be renamed (for example, you can now say `\abstract[Executive Summary]`).
- The `\keywords` macro was added to put keywords on the title page.
- The `\and` macro used in the argument to `\author` is now defined locally, instead of globally, so that you can define `\and` in your preamble to mean & or ∧.
- The vertical glue on the title page can be shrunk more, so that long abstracts can be handled more easily.

Figure and table captions are now set slightly narrower than the main body of the text, to make them visually more distinctive. Some style files also reduce the font size and the interline spacing, but `ucsc-report` does not (except in the `ieee` option).

The old `\makeindex` macro has been eliminated, and has been replaced by the `index` option in `\documentstyle`. The `\indexsubtt` macro was also eliminated as being too specialized. You can get the effect of `\indexsubtt{x}{y}` with `\indexsub{x*y}{\type{y}}`. There are now two commands for putting headings into the index without page numbers, namely `\indexonenopage` and `\indextwonopage`. The error reporting for mismatched text with the same key has been improved, so that it won't blow up, even if one of the pieces of text includes the `\verb` command.

Two new macros, `\type{x}` and `\caps{x}` have been introduced as syntactic sugar for `{\tt x}` and `{\sc x}`.

The width of marginal notes has been increased slightly, so that they almost fill the outside margin. It is a good idea to use `\raggedright` inside each `\marginpar`, as the very narrow notes cannot be decently justified.

Most of the various dimension and glue specifications in the style file have been given in terms of `ex` or `em` units, so that they scale appropriately for different font sizes. Vertical dimensions consistently use `ex`, and horizontal ones consistently use `em`.

The `\tolerance` and `\hbadness` parameters have been increased in two-column mode, to reflect the greater difficulty of justifying narrow columns. The rather high values used in the style file can result in loosely set lines, but reduce significantly the number of underfull and overfull hboxes. If you still get underfull hboxes, you might want to try `\raggedright` setting. Adding optional hyphens (`\-`) will usually eliminate any overfull hboxes.

Several parameters were modified to allow multiple floating figures without hideous page breaks. The previous settings, which were copied from the `report` style, often forced widow lines and other bad page breaks. The modified parameters are

- `\topfraction`,
- `\dbltopfraction`,
- `\floatpagefraction`,
- `\dblfloatpagefraction`,
- `\topnumber`,
- `\topfraction`,
- `\bottomfraction`, and
- `\textfraction`.

The `draft` option now puts a date on every page of a draft copy.

The new `\TODO` macro provides a convenient way of leaving reminders about uncompleted sections in draft copies. It takes a single argument, which it prints in italics as a separate paragraph. It also leaves three asterisks in the margin to show where the uncompleted material is.

The `\baselinestretch` mechanism used by LaTeX has been fixed in `ucsc-report` to leave the stretch and shrink of `\baselineskip` alone. Lamport's version discarded the stretch and shrink components, which often made it impossible to achieve flush bottoms to the pages.

Index