LaTeX2HTML Authors' Guide

Alan Bridle
National Radio Astronomy Observatory
Charlottesville
Virginia 22903

June 8, 2001

HTML Version Available[1]

[1] http://www.cv.nrao.edu/~abridle/l2h4nrao/l2h4nrao.shtml

Contents

1 Purpose 2

2 LaTeX, HTML, and PostScript/PDF 2

3 LaTeX2HTML basics 4

4 Installation and Configuration 4

4.1 Toolkit 5

4.2 Unix/Linux 5

4.3 MS-Windows 6

4.4 Initialization file 6

4.5 NRAO-specific modifications to LaTeX2HTML 7

5 General Practice 8

6 `html.sty`: hypertext extensions to LaTeX 9

7 Style Guide 10

7.1 Document header 10

7.2 Internal labels 11

7.3 External labels 11

7.4 Section titles 12

7.5 URL maintenance 12

7.6 Home button 12

7.7 Text 13

7.8 Printable version 13

8 Problems and/or Bugs 13

8.1 Superscript and subscript alignment 13

8.2 Citations 13

8.3 Definitions 14

8.4 Environments 14

8.5 In-lined images 14

8.6 Math mode font changes 14

8.7 Tolerance 14

9 Templates 15

9.1 Flexible inclusion of figures in different graphics formats 15

9.2 Separate handling of `.gif/.jpg/.png` and `.ps` input formats 16

9.3 Equation Template 17

9.4 Table Template 17

9.5 Sample Document Preamble 18

10 Sample initialization file 20

Index 25

1 Purpose

This document is a guide for NRAO authors of technical documents who want to

1. convert LaTeX documents into HTML for display on the World-Wide-Web or the NRAO Intranet, or
2. maintain both PostScript/PDF and HTML versions of a document from one LaTeX master file.

This document is not meant as a substitute for the LaTeX2HTML User Manual[2] by Drakos & Moore. Its purpose is to review some strengths and weaknesses of the LaTeX2HTML converter and to offer hints, bug work-arounds, and some style guidelines for HTML, Postscript and PDF document co-mastering in the NRAO environment.

This is a "live" document, under continual review. If you are reading it on paper, or as HTML from a site other than `www.cv.nrao.edu`, please look for updates in the master HTML version[3].

2 LaTeX, HTML, and PostScript/PDF

It is attractive to generate HTML webs automatically from LaTeX source files because

1. any valid LaTeX document is structurally robust, and
2. its basic structure (as opposed to its detailed format) can be mapped uniquely into HTML, as illustrated in Table 1.

LaTeX2HTML provides such a mapping. It uses the sectioning markup in a LaTeX source file to break the document into a cluster of HTML pages, or "nodes". While doing so, it adds hypertext navigation links based on any internal labeling (cross-references within the document). It can also build hypertext equivalents to the table of contents, indexing, and bibliography options in the master. LaTeX2HTML sorts any LaTeX items that cannot be mapped into HTML, such as equations or non-Latin symbols, into bitmapped graphics (or, optionally, PostScript) so that they can also be displayed by WWW browsers.

LaTeX2HTML also provides an `html.sty` style that adds ways to specify hypertext links to other documents, and ways to specify conditional text so that *one* LaTeX source file can specify a technical document appropriately for publication either

1. as a paper document,

[2] http://www-texdev.mpce.mq.edu.au/l2h/docs/manual/

[3] http://www.cv.nrao.edu/~abridle/l2h4nrao/l2h4nrao.shtml

Table 1: LaTeX/HTML Equivalencies

LaTeX	**HTML**
`\chapter`	<H1>
`\section`	<H2>
`\subsection`	<H3>
`\subsubsection`	<H4>
`\par`	<P>
`\begin{description}`	<DL>
`\begin{enumerate}`	<OL>
`\begin{itemize}`	<UL>
`\item`	<LI>
`\begin{table}`	<TABLE>
`\begin{figure}`	<IMG>
`\emph{text}`	<EM>text</EM>
`\textit{text}`	<I>text</I>
`\textbf{text}`	<B>text</B>
`\texttt{text}`	<TT>text</TT>
`\verb|text|`	<PRE>text</PRE>
`\label{text}`	<A NAME=”text”>
`\ref{text}`	<A HREF=”#text”>

2. as a single PostScript or PDF file, using either `LaTeX2e` and `dvips`, or `pdflatex`, or

3. as a navigable HTML file cluster or "web", using LaTeX`2HTML`.

A further benefit of using `html.sty` with `pdflatex` is that the output `.pdf` file produced by `pdflatex` can provide active links to other documents when viewed in a network-aware reader, whether or not you also create a full-fledged HTML web from the source it using LaTeX`2HTML`.

Early versions of LaTeX`2HTML` left much to be desired. They did not handle large documents reliably and their default output of symbols and equations was ugly. The "2k" implementation, while still formally a beta distribution, is more capable and robust. LaTeX`2HTML` is now being maintained and documented[4] by an open-source consortium. The documentation is comprehensive (although it lags behind the current version), and there is an extensive archived email forum[5]. The forum discusses current problems and development issues, and offers free (but often well-informed) help from other users of the package.

[4] http://www-texdev.mpce.mq.edu.au/l2h/docs/manual/

[5] http://www.xray.mpe.mpg.de/mailing-lists/latex2html/

3 LaTeX2HTML basics

LaTeX`2HTML` is a `perl` script, originated by Nikos Drakos[6] at the University of Leeds, U.K., and extended by an *ad hoc* consortium of technical documentation writers and `perl` wizards, particularly Ross Moore of the Mathematic Department at MacQuarie University in Sydney, Australia. It provides the following major capabilities:

- the document is broken into components as specified by the user and by LaTeX sectioning commands, including footnotes, tables of contents, indexing and bibliography;
- the parts of the document that LaTeX`2HTML` recognizes are converted to HTML structures, and written out, with appropriate hypertext navigation aids, as `.html` files;
- all other parts, including math symbols and equations, are passed to LaTeX to be turned into `.dvi` files, `.ps` files and finally into bitmapped `.gif` images that are cross-linked to the `.html` as appropriate;
- tables and figures are converted to HTML Table structures and/or images (depending on the HTML output level specified by the user);
- LaTeX cross-references (labels) are converted into internal hyperlinks;
- external hyperlinks can be generated, so that the `.html` files can be optimized as part of a larger document structure on the WWW or Intranet;
- conditional text structures allow the output to be optimized differently for the `.dvi` (hence PostScript) and `.html` versions;

Two basic limitations must be lived with, however:

- The converter still has some problems with LaTeX and classic TeX constructs that predate LaTeX2e. It is most reliable for documents written using *entirely* t LaTeX2e syntax. Conversion of legacy documents can be onerous, particularly for those written with clever TeX macros outside LaTeX;
- The use of inlined images to represent symbols and equations means that the HTML product is not fully resizable or fully searchable. The results may look ugly in *some* browser setups, although they are acceptable (even attractive) under a wide range of browser configurations;

4 Installation and Configuration

This section deals with installation and user-specific configuration options for LaTeX`2HTML`.

[6] http://cbl.leeds.ac.uk/nikos/personal.html

4.1 Toolkit

LaTeX2HTML requires access to the following tools, which must be installed separately:

- LaTeX2e, provided by a standard TeX installation
- perl, in a version later than 5.0
- DBM or NDBM, Unix DataBase Management
- dvips or dvipsk
- gs, Ghostscript
- pbmplus and giftrans, or the netpbm libraries (required for .gif or .png construction)

The paths to these utilities are specified at installation time by editing the perl scripts and/or the configuration file. LaTeX2HTML may be run, but without image generation (and thus without math display capability) without gs or the pbm image-translation packages.

4.2 Unix/Linux

LaTeX2HTML is normally installed in these operating systems by NRAO systems administration staff.

In some Charlottesville installations, users have needed to specify the path so the ucb library at login time, e.g. for the csh shell:

```
setenv LD_LIBRARY_PATH /usr/ucblib
```

The configuration file is latex2html.config, and the system-wide initialization file is .latex2html-init. The settings in the latter can be over-ridden by a local copy in the user's .tex document directory.

The quality of the image-generation for mathematical symbols can be improved by editing .latex2html-init to set

```
$PK_GENERATION = 1;
$METAFONT_DPI = 180;
```

This causes custom generation of the PK fonts when making .gif images of math mode expressions. This option significantly increases runing time for the conversion, but it makes much better-formed characters in the in-lined images. To enable the PK font generation, it may also be necessary to edit the MakeTexPK system script to enable font generation at 180 dpi. In Charlottesville a fix that was needed to earlier versions of MakeTeXPK was to add the 180) line as shown below:

```
if test -z "$MODE" || test "$MODE" = default; then
  case "$BDPI" in
    85) MODE=sun;;
   180) MODE=toshiba;;
   300) MODE=cx;;
   600) MODE=ljfour;;
  1270) MODE=linolo;;
```

This could be done only by systems administrators.

4.3 MS-Windows

LaTeX2HTML and all of the utilities needed to run it are now available in versions for Microsoft Windows. Detailed installation instructions for LaTeX2HTML and the associated utilities in Windows systems are available here[7].

The configuration file is `config.bat` and the system-wide settings after installation are in `l2hconf.pm` in the root directory for the LaTeX2HTML package. Line 187 of this file specifies the name for local (user or document specific) initialization files. I find it convenient to specify

```
$INIT_FILE_NAME = 'l2h.ini';
```

so that the initialization files have a file extension recognized as such by default under Windows.

4.4 Initialization file

Considerable customization can be done with the LaTeX2HTML "initialization" file. On Unix systems this will be called `.latex2html-init`; on Windows systems it is more convenient to name it `l2h.ini` (see Section 4.3. The master version provided with LaTeX2HTML can be copied to the LaTeX2HTML `.tex` source directory of any user, and customized there by editing with any text editor.

Options specified in a per-user, or per-document, copy of the initialization file over-ride those in the system-wide copy. This file is therefore a convenient tool for tailoring LaTeX2HTML settings to suit the needs of a particular document or user.

The `$ADDRESS` line in the initialization file can be modified to provide author contact information at the foot of every HTML page. It will default to your computer login name, but the following example shows how to invoke the NRAO-standard contact information tool. Note the use of `perl` syntax for embedded quote marks.

[7] http://www.cv.nrao.edu/~abridle/toolmemo/toolmemo.shtml

```
$ADDRESS = "
<DIV ALIGN=left>
<FORM action=\"http://www.cv.nrao.edu/cgi-bin/contact\" method=\"post\">
<INPUT type=\"hidden\" name=\"key\" value=\"Alan+Bridle\">
<INPUT type=\"submit\" value=\"Alan Bridle\"><BR>
<STRONG>\n$address_data[1]</STRONG></FORM></DIV>
";
```

A copyright statement could also be added to each page in this fashion.

Other global variables that can also be set usefully from the initialization file include:

`$SHOW_SECTION_NUMBERS = 1` will cause section numbers to be shown in the HTML version, matching those in the LaTeX version. Hiding the section numbers permits use of sections as stand-alone web documents, but showing them helps readers to navigate a complex web by reminding them of its structure. If section numbers are hidden, the HTML cross reference to a section is shown using a default symbol, rather than the section's number.

`$WORDS_IN_PAGE = 150` replicates the navigation button panel at the bottom of any page containing more than 150 words (making no allowance for in-lined images or preformatted text). value, is useful. This speeds navigation by minimizing the scrolling that a reader must do to reach the button panel. The default (of 200) here seems a little over-generous.

`$WORDS_IN_NAVIGATION_PANEL_TITLES = 8` enlarges the section-title fragments that are used in the text version of the navigation button panel. The default value (of 4) makes the text navigation panel rather terse and possibly ambiguous.

`$MATH_SCALE_FACTOR` and `$FIGURE_SCALE_FACTOR` offer control over the global scalings of in-line images and figures, respectively. The defaults (1.6) are sensibly chosen but may be inappropriate for specific applications. (Figures may also be scaled individually as they are converted to `.gif` images using the arguments of the `htmlimage` command in the LaTeX source file.)

`$PK_GENERATION = 1;` will invoke custom generation of the PK fonts when making `.gif` representations of symbols. This produces much higher-quality output than the `$PK_GENERATION = 0;`, and should probably be used as the system-wide default. If the system-wide option has been turned off to speed up the conversion, you may want to set the custom generation flag locally, or negotiate the option with your system administrator. I also recommend setting `$ANTI_ALIAS_TEXT = 0;` to turn off anti-aliasing while generating images of typeset symbols and equations with the usual font sizes used in LaTeX documents, as this option works well only with very large fonts.

4.5 NRAO-specific modifications to LaTeX2HTML

When using LaTeX`2HTML` for documents that will be published on the NRAO web pages, it is useful if the HTML output incorporates the standard NRAO server-side includes so that it conforms to the observatory's uniform web style. If you obtained your copy of LaTeX`2HTML` from the NRAO distribution on `Cvsnap1`, it contained the option to do this, which you can exercise by setting the variables

```
$NRAO = 1;
$ALLOW_SSI = 1;
```

in the initialization file. The first setting activates the NRAO options, and the second tells LaTeX2HTML to write out the files with .shtml extensions that will invoke the server-side includes when installed on an NRAO web server.

Setting

```
$NRAO = 0;
$ALLOW_SSI = 0;
```

in the initialization file will produce the default (i.e., non-NRAO-specific) LaTeX2HTML output.

If you did not obtain your LaTeX2HTML from Cvsnap1, you can add the NRAO options to it by hand-editing the master perl script (latex2html.bat in the Windows version) exactly as described[8] for latex2html.pin in *TeX, LaTeX, and HTML Tools for Windows PC's* (the line numbers to be edited will not be identical to the example shown there, but the required perl code changes are exactly the same).

5 General Practice

After customizing .latex2html-init (Section 4.4) as needed, LaTeX2HTML can be run on filename.tex with the commands:

```
latex2html filename
```

or

```
latex2html -html_version n filename
```

where n specifies the HTML standard level of the .html output.

By default, LaTeX2HTML writes the .html and .gif output to a directory called filename one level below the source .tex file, and the title HTML page is named filename.html. The output can be redirected to another directory using the -dir command at the command line.

It is important to run LaTeX2e, makeindex (if indexing the document), and LaTeX2e again on the .tex file immediately before running LaTeX2HTML. Successful execution of LaTeX2e ensures that LaTeX2HTML finds an appropriate .aux file without warning messages about referencing, etc. This guards against missing or obsolete internal references, whose corresponding hypertext links can be hard to find and debug once they are embedded in the .html output.

[8] http://www.cv.nrao.edu/~abridle/toolmemo/node29.shtml

6 html.sty: hypertext extensions to LaTeX

LaTeX2HTML provides the file `html.sty` which defines extensions to standard LaTeX that are important when co-mastering Postscript and HTML documents.

`html.sty` **provides three new environments**, invoked by

```
\begin{environment_name}
....
\end{environment_name}
```

htmlonly An environment within which LaTeX commands will be interpreted only when producing the HTML output with LaTeX2HTML . Its contents are ignored by `LaTeX2e` when generating the `.dvi`, hence PostScript, output.

latexonly An environment within which LaTeX commands will be interpreted only when producing the `.dvi` output. Its contents are ignored when generating the `.html` output with LaTeX2HTML .

rawhtml An environment within which you can insert HTML commands directly for processing when LaTeX2HTML generates the `.html` output. Its contents are ignore by t LaTeX2e when generating the `.dvi` output.

`html.sty` **also provides eight new commands:**

htmlref A direct way to make an *internal* hypertext link to a LaTeX `\label{reference}` statement. The syntax is `\htmlref{text}{reference}`. The `.dvi` output will show only the `text` string. The `.html` will highlight `text` as a internal hypertext link to the place in the document where `\label{reference}` marked a section, equation, table, etc.

htmladdnormallink As above, except that the hypertext link is to an *external* URL, e.g. to: `http://www.cv.nrao.edu/cv-home.html`.

htmladdnormallinkfoot As above, except that in the `.dvi` output the URL of the link is displayed as a footnote. (For an example of its use, see Page 4 of this document.)

hyperref A conditional text command. `\hyperref{text1}{text2}{text3}{reference}` displays `text1` as the highlit text in the `.html` output, with an internal hypertext link to the `reference` label. In the `.dvi` output, `text2` is the prefix text for, and `text3` the suffix text to, the counter value corresponding to `reference`. For example,

```
\hyperref{Style Guide}{Style Guide (see Section}{below)}{sec:style}
```

produces the text

```
Style Guide (see Section 7 below)
```

in the `.dvi` output and a `Style Guide` hyperlink in the `.html` output.

htmlimage Used inside any environment that is converted into an inlined image (e.g., a `figure` environment) to control how the image will be translated when LaTeX2HTML is run. The arguments are a string of options separated by commas:
`[scale=<scale factor>],[external],[thumbnail=<reduction factor>].`
The `scale` option controls the size of the final image. The `external` causes the image not to be inlined (the default), but instead to be made accessible via a hypertext link. The `thumbnail` option causes a small inlined image to be placed in the caption; the thumbnail size depends on the reduction factor (use of `thumbnail` implies use of `external`). For example: `\htmlimage{scale=1.5,external,thumbnail=0.2}` causes a thumbnail image 1/5th of the original size to be placed in the document, pointing to an external image 1.5 times bigger than the original.

htmladdimg The syntax is `\htmladdimg{URL}`, where the single argument `URL` specifies the URL of an image that will be inserted only in the `.html` output.

htmlcite provides a reference in the HTML version only. First argument is text for both versions, second is citation for use in the HTML version.

htmlrule Adds a horizontal rule, valid even within a figure caption.

7 Style Guide

This Section proposes style guidelines for LaTeX authors who wish to display documents on the Web or an Intranet.

7.1 Document header

A sample syntax for the header of a document using:

1. `html.sty` for the LaTeX2HTML hypertext extensions,
2. `makeidx.sty` for index generation using LaTeX `\index` tags, and
3. `epsf.sty` for inserting Postscript files as graphics.

is:

```
\documentclass{article}
\usepackage{html,makeidx,epsf}
```

Do not use the older LaTeX `\documentstyle` command to call up style files. Many bugs emerge from LaTeX2HTML when this is done!

Section 9.5 below gives a suggested template for the preamble of a master file for NRAO documents.

7.2 Internal labels

Mark every section, figure, equation and table with a `\label`, e.g. `\label{fig:text}` for a figure. Then refer to it by this label, e.g. `Figure~\ref{fig:text}`. Such cross-referencing serves two important purposes. First, it automates numbering throughout the `.dvi`, hence PostScript, output. Second, it creates internal hypertext links through the `.html` output, making the `.html` rendition easier to navigate.

A good style for the label texts is `sec:text` for sections, `sub:text` for subsections, `fig:text` for figures, `eqn:text` for equations, `tbl:text` for tables, etc. This style speeds searches for all labels of one kind when editing the file, and helps document maintenance.

Note that the `\label` command must come after, or within, the `\caption` command for Figures and Tables, in order to generate the correct reference.

7.3 External labels

LaTeX2HTML has a way to extend the label cross-referencing across documents. The list of label references is stored in the same directory as the `.html` and `.gif` output in a `perl` file called `labels.pl`. One document can cross-reference another's labels via the `\externallabels` command, as in the following example:

```
\externallabels{Coding_URL}%
               {/aips++/docs/reference/Coding/labels.pl}
\externallabels{HowTos_URL}%
               {/aips++/docs/reference/HowTos/labels.pl}
\externallabels{System_URL}%
               {/aips++/docs/reference/System/labels.pl}
```

The first argument is the URL to the directory containing the external document; the second is the full pathname to the `labels.pl` file containing the external document. Hypertext links can then be made to any of the documents whose label files have been imported in this way, using the command:

```
\htmlref{label_name_used_in_external_document}
```

This approach has the merit that if the `labels.pl` files in the external documents become expanded (but their directory locations stay the same), hypertext references in the documents that call them can be updated simply by rerunning LaTeX2HTML on the calling documents.

To take full advantage of external labeling in a complex documentation package, the label-naming scheme might be extended to `doc-label:fig:text` for figures, `doc-label:eqn:text` for equations, etc., wherein the `doc-label` strings distinguish entire documents.

7.4 Section titles

Do not use math mode in `\section` or `\subsection` titles: LaTeX2HTML has to pipe math mode quantities to LaTeX when converting them into in-line images. It does not do this within section titles, so any math mode quantities that such titles contain are ignored in the `.html` output.

7.5 URL maintenance

Gather all external hyperlinks into definitions at the start of the LaTeX file. This helps to maintain the document when hyperlinks change, especially if multiple instances of the same hyperlink text occur (as they will if the printed document shows plain text of the links; or if the same link is made several times). It should also make it easier to gather these hyperlinks into communal input files for easier long-term maintenance. e.g.

```
\htmladdnormallink{{\it VLA Calibrator List}}{\vlacaliblink}
```

is easier to adjust throughout a document if the hyperlink alters, by changing the definition for `vlacaliblink`:

```
\def\vlacaliblink{http://www.nrao.edu/\~{}gtaylor/calib.html}
```

Note the work-around in the above example for the tilde in a URL (the tilde will otherwise be interpreted in TeX mode as a space, breaking the link). The tilde could also be inserted in `rawhtml` mode.

7.6 Home button

Add a `Home` button to the LaTeX2HTML navigation panel. It is impolite to trap a reader inside a complex hypertext document with no built-in escape to the main documentation Web. A sample syntax for adding a `Home` button that provides a more graceful escape to your home page is:

```
\htmladdtonavigation{\htmladdnormallink
  {\htmladdimg{../home.gif}}{http://www.cv.nrao.edu/\~{}abridle/}}
```

This example (which could be placed anywhere in the document preamble) puts a button image `home.gif`, in this case from the directory above the `.html` output), into every navigation panel throughout the `.html` file cluster. In the example, the button is linked to my personal home page: substitute other links as appropriate! I have made a `home.gif` image that matches the default LaTeX2HTML navigation-button style. You can copy a button from the version[9] of this document on the NRAO Web for use with other LaTeX2HTML-converted documents.

If you use the standard NRAO server-side includes, every page will automatically get the usual NRAO-wide navigation bar, but this will not include a link to your own home page.

[9] http://www.cv.nrao.edu/~abridle/l2h4nrao/l2h4nrao.shtml

7.7 Text

Avoid text that makes sense only in the Postscript or in the HTML version (“above”, “below”, “click here”, etc.), except in contexts that are themselves version-specific (in `latexonly` environments, in `hyperref` fields, etc.). “Above” and “below” may not make sense in the hypertext version, where the material referred to may appear on other hypertext “pages”. Equally, mouse-oriented language such as “click here” should not appear in the PostScript version.

7.8 Printable version

Put a hypertext link to the Postscript (or compressed-Postscript) or PDF version of a complex document near the top of its first HTML page. This makes it easier to obtain hard copy, and many readers may also prefer to navigate a complex document on paper, rather than on a screen. Include an estimate of the size of the Postscript, to warn the reader about load-time. The HTML template in Section 9.5 below gives an example of the suggested style.)

8 Problems and/or Bugs

There are a few bugs, especially when converting legacy documents that were not writted for LaTeX2e, but there are known work-arounds for many of them.

8.1 Superscript and subscript alignment

LaTeX2HTML cannot guarantee proper alignment between superscripts and subscripts and their parent characters using math mode unless the parents are contained in the same math mode call as the superscript or subscript.

For example, use `$\mathrm{P}^2$`, not `P$^2$`, to produce P$^2$.

`P$^2$` will write P$^2$ with a small 2 inline, not a superscript 2 as needed. (This is one of the most common flaws in documents converted with LaTeX2HTML, but may be addressed better once use of the `<SUB>` and `<SUP>` tags in browsers becomes widespread.)

Similarly, do not use `10$^{\circ}$` for $10^{\circ}$, but put the number of degrees inside the same math mode call, i.e. `$10^{\circ}$`.

8.2 Citations

`\hypercite` creates a textual link to the bibliography page where citation details are shown and is recommended as a substitute for the `\cite` command in LaTeX.

`\htmlcite` creates a textual link to the bibliography page without showing details in the LaTeX version.

8.3 Definitions

You cannot use multiple definitions based on the same `\def`, `\newcommand` in a document to be processed by LaTeX2HTML ; only the last-given definition will be used throughout the whole document.

Do not use a backslash to create blank space after using a `\def` macro in your text: the `.html` will show the backslash, not the desired space! One work-around for this is to follow such constructs with a tilde in the text. Another is to define the macro in versions both with and without trailing space (for use within text and before punctuation marks, respectively).

8.4 Environments

Because LaTeX2HTML processes document sections as independent units, environments cannot cross section boundaries.

8.5 In-lined images

The running time of LaTeX2HTML can be greatly reduced by re-using images from previous runs (which is its default).

It is essential to look carefully at the output of any LaTeX2HTML run that gives error messages about image generation and conversion. If you see any errors about missing files in the image-conversion section, LaTeX2HTML may not have assigned the right output images to symbols or equations. While I have not seen this failure mode in any LaTeX source file that is syntactically correct, LaTeX2HTML is less forgiving than LaTeX itself about minor syntax errors. If you encounter image conversion errors, it is essential to trace their source and rectify it. Note the number of the image that failed to generate correctly, and inspect the `images.log` and `images.tex` files in the LaTeX2HTML *output* directory to locate and correct the `.tex` source text that produced the error.

8.6 Math mode font changes

Math mode font changes made outside the math mode are not honored. Thus the two equations in `$a_b$` and `{\LARGE $a_b$}` would come out looking the same. The trick is to write `$\mbox{\LARGE $a_b$}$`.

8.7 Tolerance

TeX `tolerance` commands are useful for temporarily relaxing white-space restrictions to get around horizontal overflows. But use them only inside `latexonly` environments, else their numerical arguments will appear in the `.html` output!

9 Templates

This section suggests some templates for commonly-used elements of document.

9.1 Flexible inclusion of figures in different graphics formats

Unfortunately, no single graphics input format is supported by *all three* of LaTeX, `pdflatex`, and LaTeX2HTML.

Table 2: Graphics Support

Package	**Input Format**	**Output Format**
LaTeX	`.eps, .ps`	`.dvi, .ps`
`pdflatex`	`.jpg, .pdf, .png, .tif`	`.pdf`
LaTeX2HTML	same as LaTeX	`.gif/.png, .html`

This complicates production of graphics-rich documents if Postscript, PDF and HTML output are all required. It *is* possible however to code a LaTeX `figure` environment call so that the same source text brings in different graphics files as needed by the different compilers. The following template shows how to do this with `\usepackage{graphicx}` included in the document preamble:

```
\begin{figure}[thp]
\begin{center}
\includegraphics[width=4in]{fig1}
\caption{Caption text.}
\label{fig:labeltext}
\end{center}
\end{figure}
```

`\includegraphics` will search for the file `fig1.eps` when LaTeX is run, either directly or by running LaTeX2HTML. LaTeX2HTML will convert the input `fig1.eps` graphic into a `.gif` or `.png` file using the LaTeX engine, then the `dvips`, `Ghostscript` and `netpbm` utilities. The same `\includegraphics` call will also search for whichever of `fig1.pdf` or `fig1.jpg` are in the source directory if `pdflatex` is run to produce `.pdf` output.

The example shown requests that the graphic be reproduced with a width of 4 inches. `includegraphics` also lets you specify a scale factor, angle of rotation, image width, and/or image height, for example:

```
\includegraphics[scale=0.5]{<filename>}
```

```
\includegraphics[angle=45]{<filename>}
\includegraphics[width=2in]{<filename>}
\includegraphics[totalheight=4in]{<filename>}
\includegraphics[scale=0.5,totalheight=4in]{<filename>}
```

9.2 Separate handling of .gif/.jpg/.png and .ps input formats

This template includes a JPG file `fig1.jpg` of width `xxx` pixels and height `yyy` pixels directly in the HTML output, and the same figure as a PostScript file `fig1.ps` in the PostScript output with height 7.5 in, assuming the older `\usepackage{epsfig}` in the document preamble. The two equivalent versions of `fig1` would in this case be prepared separately outside LaTeX2HTML.

```
\begin{rawhtml}
<div align="center">
<img src="fig1.jpg" width="xxx" height="yyy" alt="Alt text"><BR>
<small>Caption text</small>
</div>
\end{rawhtml}

\begin{figure}
\begin{center}

%uncomment for actual use --
%\begin{latexonly}
%\epsfysize=7.5in
%\epsfbox{fig1.ps}
%\end{latexonly}

\caption{Caption text.\label{fig:example}}
\end{center}
\end{figure}
```

In this example, each type of graphics input is sent to the corresponding output type independently by using conditional (`rawhtml, latexonly`) text environments within LaTeX2HTML. The first simply writes the code needed to include the `.jpg` image directly in the HTML, while the second gives the TeX instructions to produce the Postscript version.

To use a pre-prepared `fig1.gif` or `fig1.png` in the same way, substitute `fig1.gif` or `fig1.png` for `fig1.jpg` in this example.

9.3 Equation Template

```
\begin{center}
\begin{equation}
\label{eqn:example}

<TeX code for equation>

\end{equation}
\end{center}
```

The equation can then be referred (and hyperlinked) to elsewhere in the text by `Equation~\ref{eqn:example}`.

9.4 Table Template

A simple 2 × 2 table, with caption and label.

```
\begin{table}
\caption{Caption text}
\label{tbl:example}
\begin{center}
\begin{tabular}{|c|c|}
\hline
\textbf{Column 1}  & \textbf{Column 2} \\
\hline
data[1,1] & data[1,2] \\
data[2,1] & data[2,2] \\
\hline
\end{tabular}
\end{center}
\end{table}
```

When composing tables in TeX for conversion to HTML, authors must be aware of a small difference between TeX and HTML that LaTeX2HTML must mediate. TeX does not delineate row borders unless the user supplies an `\hline` whereas HTML applies the border information for a Table to every row in the table. TeX also interprets `\\` inside the `tabular` environment format as an unuequivocal start of a new row, equivalent to a `<TR>` in HTML. When writing a long table entry that may extend over several lines, you cannot therefore use `\\` as a line break within a table row, as in TeX because this will be interpreted as a new table row (and separated by a border) in the HTML version.

The workaround for this difference between TEX and HTML treatment of table borders is to use paragraph (p) format for table columns that will contain cells with lengthy text, as in the following example.

```
\begin{tabular}{|c|p{3in}|}
\hline
\TeX\  &  A publication-quality typesetting program that provides
          complete control of formatting for printed documents, with
          particularly strong capabilities for displaying mathematical
          equations and formulae..\\
\hline
HTML   &  A content markup language developed for document
          display on the World Wide Web, leaving many aspects of
          format control to the reader's web browser settings.
\hline
\end{tabular}
```

9.5 Sample Document Preamble

The following is an example of a LATEX document preamble for use with LATEX2HTML. It is based on this Guide's preamble.

```
% N.B. one {latexonly} environment commented out so that its
% contents can be displayed in the HTML version of this template.
% Uncomment it for actual use!
%
% Use text editor to replace:
%
%       author   --- author's login name
%       thisdoc  --- document filename (as in thisdoc.tex, thisdoc.ps)
%       psiz     --- size of compressed PostScript file
%
%       Document_Date          --- current date
%       Document_Short_Title   --- header text for Postscript
%       Document_Long_Title    --- full document title
%       Author_Name            --- full author name
%       Author_City            --- Charlottesville, Socorro, etc.
%       Author_State           --- Virginia, New Mexico, etc.
%
%       (non-NRAO: also replace institute name/acronym and country?)
%

\documentclass{article}
```

```
\usepackage{html,makeidx,epsf}

%
% Add home page navigation button -- edit the URL!
%


\htmladdtonavigation{\htmladdnormallink
  {\htmladdimg{../home.gif}}{http://www.cv.nrao.edu/\~{}author/}}

\begin{latexonly}
\tolerance=200
\pagestyle{headings}
\textheight  8.5truein
\textwidth   6truein
\topmargin   0.25truein
\oddsidemargin 0.25truein
\evensidemargin 0.25truein
\end{latexonly}

%
% define hyperlink URLs:
%

\def\authorhomepage{http://www.cv.nrao.edu/\~{}author/}
\def\cvhomelink{http://www.cv.nrao.edu/cv-home.html}
\def\thisdocURL{http://www.cv.nrao.edu/\~{}author/thisdoc/thisdoc.html}
\def\vlahomelink{http://www.nrao.edu/doc/vla/html/VLAhome.shtml}

\makeindex

\begin{document}

%
%  Page formatting for Postscript output
%

\title{
{\bf Document_Long_Title}
}

\author
{
Author_Name\\
National Radio Astronomy Observatory\\
Author_City, Author_State, U.S.A.
}
```

```
\date
{
{Text last updated: Document_Date}\\
}

\begin{center}
\begin{latexonly}
\htmladdnormallinkfoot{HTML Version Available}{\thisdocURL}
\end{latexonly}
\begin{htmlonly}
\htmladdnormallink{PDF Version}{\thisdocPDF}
\end{htmlonly}
\end{center}

% uncomment to run: \begin{latexonly}
\markright{Document_Short_Title}
\maketitle
% uncomment to run: \end{latexonly}

\tableofcontents

\pagebreak

%
% optional post-title formatting for PostScript
%
\parindent0pt
\parskip2.5ex plus 0.5ex minus 0.5ex
```

10 Sample initialization file

```
#LaTeX2HTML Version 2k initialization
#
# modified by AHB for white bgrd, abridle address lookup, PK font generation
#
### Command Line Argument Defaults #######################################

$MAX_SPLIT_DEPTH = 8; # Stop making separate files at this depth

$MAX_LINK_DEPTH = 4;    # Stop showing child nodes at this depth

$NOLATEX = 0;           # 1 = do not pass unknown environments to Latex

$EXTERNAL_IMAGES = 0;   # 1 = leave the images outside the document
```

```
$ASCII_MODE = 0;        # 1 = do not use any icons or internal images

# 1 =  use links to external postscript images rather than inlined GIF's.
$PS_IMAGES = 0;

# To turn on NRAO options set NRAO = 1 here
$NRAO = 1;

# To turn on server-side includes and write out .shtml
$ALLOW_SSI = 1;

$TITLE = $default_title;      # The default is "No Title"

$DESTDIR = '';         # Put the result in this directory

# When this is set, the generated HTML files will be placed in the
# current directory. If set to 0 the default behaviour is to create (or reuse)
# another file directory.
$NO_SUBDIR = 0;

# Supply your own string if you don't like the default <Name> <Date>
#
# AHB version provides the NRAO logo, a button for the NRAO contact system, and the date
#
$ADDRESS = "<DIV ALIGN=left><FORM action=\"http://www.cv.nrao.edu/cgi-bin/contact\" method=\"post\">
<INPUT type=\"hidden\" name=\"key\" value=\"Alan+Bridle\">
<INPUT type=\"submit\" value=\"Alan Bridle\"><BR><STRONG>\n$address_data[1]</STRONG></FORM></DIV>
";

$NO_NAVIGATION = 0; # 1 = do not put a navigation panel at the top of each page

# Put navigation links at the top of each  page.  If  the page  exceeds
# $WORDS_IN_PAGE  number of words then put one at the bottom of the page.
$AUTO_NAVIGATION = 1;

# Put a link to the index page in  the  navigation  panel
$INDEX_IN_NAVIGATION = 1;

# Put a link to the table of contents  in  the  navigation  panel
$CONTENTS_IN_NAVIGATION = 1;

# Put a link to the next logical page  in  the  navigation  panel
$NEXT_PAGE_IN_NAVIGATION = 1;

# Put a link to the previous logical page  in  the  navigation  panel
$PREVIOUS_PAGE_IN_NAVIGATION = 1;

$INFO = 1;              # 0 = do not make a "About this document..." section
```

```
# Reuse images generated during previous runs
$REUSE = 2;

# When this is 1, the section numbers are shown. The section numbers should
# then match those that would have bee produced by LaTeX.
# The correct section numbers are obtained from the $FILE.aux file generated
# by LaTeX.
# Hiding the seciton numbers encourages use of particular sections
# as standalone documents. In this case the cross reference to a section
# is shown using the default symbol rather than the section number.
$SHOW_SECTION_NUMBERS = 1;

### Other global variables ###############################################
$CHILDLINE = "<BR> <HR>\n";

# This is the line width measured in pixels and it is used to right justify
# equations and equation arrays;
$LINE_WIDTH = 500;

# Used in conjunction with AUTO_NAVIGATION
$WORDS_IN_PAGE = 150;

# Affects ONLY the way accents are processed
$default_language = 'english';

# The value of this variable determines how many words to use in each
# title that is added to the navigation panel (see below)
#
$WORDS_IN_NAVIGATION_PANEL_TITLES = 8;

# These settings ask for custom anti-aliased generation of symbol images
# in the text of small-font documents
$PK_GENERATION = 1;
$METAFONT_DPI = 180;
$ANTI_ALIAS_TEXT = 0;

# This number will determine the size of the equations, special characters,
# and anything which will be converted into an inlined image
# *except* "image generating environments" such as "figure", "table"
# or "minipage".
# Effective values are those greater than 0.
# Sensible values are between 0.1 - 4.
$MATH_SCALE_FACTOR = 1.5;

# This number will determine the size of
# image generating environments such as "figure", "table" or "minipage".
# Effective values are those greater than 0.
# Sensible values are between 0.1 - 4.
$FIGURE_SCALE_FACTOR = 1.5;


#  If this is set then intermediate files are left for later inspection.
#  This includes $$_images.tex and $$_images.log created during image
#  conversion.
#  Caution: Intermediate files can be *enormous*.
$DEBUG = 0;

#  If both of the following two variables are set then the "Up" button
#  of the navigation panel in the first node/page of a converted document
#  will point to $EXTERNAL_UP_LINK. $EXTERNAL_UP_TITLE should be set
#  to some text which describes this external link.
$EXTERNAL_UP_LINK = "";
$EXTERNAL_UP_TITLE = "";

# If this is set then the resulting HTML will look marginally better if viewed
# with Netscape.
$NETSCAPE_HTML = 0;

# Valid paper sizes are "letter", "legal", "a4","a3","a2" and "a0"
# Paper sizes has no effect other than in the time it takes to create inlined
# images and in whether large images can be created at all ie
#  - larger paper sizes *MAY* help with large image problems
#  - smaller paper sizes are quicker to handle
$PAPERSIZE = "a0";

# Replace "english" with another language in order to tell LaTeX2HTML that you
# want some generated section titles (eg "Table of Contents" or "References")
# to appear in a different language. Currently only "english" and "french"
# is supported but it is very easy to add your own. See the example in the
# file "latex2html.config"
$TITLES_LANGUAGE = "english";

### Navigation Panel ##########################################################
#
# The navigation panel is constructed out of buttons and section titles.
# These can be configured in any combination with arbitrary text and
# HTML tags interspersed between them.
# The buttons available are:
# $PREVIOUS - points to the previous section
# $UP  - points up to the "parent" section
# $NEXT - points to the next section
# $NEXT_GROUP - points to the next "group" section
# $PREVIOUS_GROUP - points to the previous "group" section
# $CONTENTS - points to the contents page if there is one
# $INDEX - points to the index page if there is one
```

```
#
# If the corresponding section exists the button will contain an
# active link to that section. If the corresponding section does
# not exist the button will be inactive.
#
# Also for each of the $PREVIOUS $UP $NEXT $NEXT_GROUP and $PREVIOUS_GROUP
# buttons there are equivalent $PREVIOUS_TITLE, $UP_TITLE, etc variables
# which contain the titles of their corresponding sections.
# Each title is empty if there is no corresponding section.
#
# The subroutine below constructs the navigation panels in each page.
# Feel free to mix and match buttons, titles, your own text, your logos,
# and arbitrary HTML (the "." is the Perl concatenation operator).
sub top_navigation_panel {

    # Put in a body statement (could also have text color here)
    "<BODY BGCOLOR=\"ffffff\">" .

    # Now add a few buttons with a space between them
    "$NEXT $UP $PREVIOUS $CONTENTS $INDEX $CUSTOM_BUTTONS" .

    "<BR>\n" .    # Line break

    # If ‘‘next’’ section exists, add its title to the navigation panel
    ($NEXT_TITLE ? "<B> Next:</B> $NEXT_TITLE\n" : undef) .

    # Similarly with the ‘‘up’’ title ...
    ($UP_TITLE ? "<B>Up:</B> $UP_TITLE\n" : undef) .

    # ... and the ‘‘previous’’ title
    ($PREVIOUS_TITLE ? "<B> Previous:</B> $PREVIOUS_TITLE\n" : undef) .

    #  Line Break, horizontal rule (3-d dividing line) and new paragraph
    "<BR> <P>\n"
}

sub bot_navigation_panel {

    #  Start with a horizontal rule (3-d dividing line)
    "<HR>".

    # Now add a few buttons with a space between them
    "$NEXT $UP $PREVIOUS $CONTENTS $INDEX $CUSTOM_BUTTONS" .

    "<BR>\n" .    # Line break

    # If ‘‘next’’ section exists, add its title to the navigation panel
    ($NEXT_TITLE ? "<B> Next:</B> $NEXT_TITLE\n" : undef) .
    # Similarly with the ‘‘up’’ title ...
    ($UP_TITLE ? "<B>Up:</B> $UP_TITLE\n" : undef) .

    # ... and the ‘‘previous’’ title
    ($PREVIOUS_TITLE ? "<B> Previous:</B> $PREVIOUS_TITLE\n" : undef)

}


1;  # This must be the last line
```

Index

address link, 6

background color, 6
backslash, 14
bugs, 13

cite, 13
command mapping, 2
conditional text, 2, 4, 9
configuration file, 5

DBM, 5
def, 14
degree symbol, 13
Drakos, N., 2, 4
dvips, 5

epsf.sty, 10
epsfig, 16

giftrans, 5
graphicx, 15
gs, 5

Home button, 12, 17
HTML version, 8
html.sty, 2, 9, 10
htmladdimg, 9, 12
htmladdnormallink, 9
htmladdnormallinkfoot, 9
htmlcite, 9
htmlimage, 6, 9
htmlonly, 9
htmlref, 9
htmlrule, 9
hyperref, 9, 13

images
- gif, 4–6, 8, 9, 15, 16, 20
- jpg, 15, 16
- png, 16

includegraphics, 15
indexing, 8, 17
initialization file
- customization, 6
- Unix, 5
- Windows, 6

labels, 2, 4, 9, 11
LaTeX2e, 2, 5, 8, 9, 13
LaTeX2HTML manuals, 2
latexonly, 9, 13, 14
limitations, 4
Linux, 5

makeidx.sty, 10
MakeTexPK, 6
Moore, R., 4
MS-Windows, 6

navigation panel, 4, 6, 12, 17, 20
NDBM, 5
netpbm, 5
newcommand, 14

pbmplus, 5
PDF, 2
perl, 4–6

rawhtml, 9, 12

subscripts, 13
superscripts, 13

tilde, 12, 14
tolerance, 14

Unix, 5