TeX, LaTeX and HTML Tools for Windows PC's

Alan Bridle
National Radio Astronomy Observatory
Charlottesville
Virginia 22903

April 18, 2001

HTML Version Available[1]

[1] http://www.cv.nrao.edu/~abridle/toolmemo/toolmemo.html

Contents

1 Summary

This document describes free tools that support mastering of technical documents in TEX or LATEX with output either to Postscript or PDF files, or to an HTML file cluster, on PC's running Microsoft Windows. These tools can produce a range of output formats for different publication media from the same `.tex` source, at zero cost in software.

Installation kits for all of the packages described here, including an option to emit web pages in the NRAO standard format automatically from LATEX source, are available in the NRAO NT domain from the `Cvsnap1` server.

2 TeX for the PC

Although WYSIWYG Windows "office" software is easier to use for non-technical word-processing, TEX and/or LATEX are still the typesetting systems of choice for many technical documents. This is because

- they provide a high-quality display for symbols and equations,
- they have been adopted as standards for electronic document submission to many technical publishers and preprint archives, and
- they use ASCII text format for the document source files, so that these are portable and can be archived without fear that their format will become unreadable.

Most scientists are familiar with TEX and LATEX and many also have access to PC's running MS-Windows at work and/or at home. TEX-based packages for MS-Windows are therefore likely to be of interest to authors of technical documents for some time, and several comprehensive packages have been developed. Modern publication also calls for format flexibility: output may be needed on paper, as compact single files for electronic document transmission, or as an easily-navigable web with hyperlinks to other material. Postscript, PDF and HTML file clusters, or a mixture of these, are all desirable output formats. Free software for producing all of these formats from a single TEX or LATEX source has been available for some time on the NRAO's Unix/Linux platforms. The same capabilities can also be provided, *at zero cost other than disk space and installation effort*, on the Win9x, NT or Win2k systems.

Section 2.1 describes a freely-available and well-supported TEX and LATEX package for Microsoft Windows that I have used and can recommend; Section 2.2 mentions alternatives.

Section 3 describes a set of packages that can be installed together to convert LATEX source to HTML.

2.1 MiKTeX

The open-source and widely-used `MiKTeX`[2] package is available over the net as a free download[3]. A CD-ROM version is scheduled for release in June 2001. `MiKTeX 2.0`, released in January 2001 and archived in Charlottesville at `Cvsnap1\SW\texutils\MiKTeX2`, includes:

- TEX and LATEX2*e*,
- `pdfTeX` and `pdflatex` for directly producing `.pdf` output,
- `dvips` and `dvipdfm` for converting `.dvi` to Postscript and `.pdf`,
- `Yap`, a `.dvi` previewer that can display included Postscript images,

[2] http://www.miktex.org/
[3] http://www.miktex.org/latest.html

- many commonly used utilities, e.g. `BibTeX`, `MakeIndex`, `TeXinfo`, etc.,
- a complete set of public-domain Type-1 (PostScript) fonts for the Computer Modern, LaTeX and AMS math families,
- a Windows interface to help you configure the package,
- optional enhanced TeX features and a TeX-oriented editor,
- `texify`, a TeX compiler driver that will run LaTeX, `pdflatex`, `MakeIndex` and `BibTeX` as many times as are needed to produce a `.dvi` or `.pdf` file with sorted indices and all of the cross-references resolved.

2.1.1 Installation

`MiKTeX` installation is easy using a setup wizard supplied with the package. Do not, however, use the default installation folder name of `Program Files\MiKTeX`, or any other folder name that embeds a space, if you will also use LaTeX2`HTML` with `MiKTeX`. I suggest that you instead tell `MiKTeX` to install under `D:\texutils`. You can then use the path names in Section A (and will be able to install the NRAO variant of LaTeX2`HTML` without editing any configuration scripts).

`MiKTeX` does not require administrator privilege to instal, but on an NT system you should login as an administrator so that you can modify the `path` to make the `tex` and `latex` commands available while working in a document directory. For an installation on drive `D:`, you would add `D:\texutils\miktex\bin\;` to the path, for example.

Other configuration options can be set up from `MiKTeX`'s own Windows interface at `Start|Programs|MikTeX2|MiKTeX Options`.

The installed size of `MiKTeX`, including all common fonts and utilities but not the source code, is 73 Mbytes.

To uninstall `MiKTeX`, use `Start|Settings|Control Panel|`, then click on `Add/Remove programs` and again on `MiKTeX 2` in the program list. The uninstaller will undo everything except the `path` changes.

2.1.2 Use and documentation notes

- `MiKTeX` runs in the MSDOS-like Command Prompt window, but you can preview `.dvi` files with `Yap` by clicking on them in an *Explorer* file management window.
- `Yap` can print from `.dvi` directly to Postscript printers but the printout is slow, and does not include page numbering. It is better to run `dvips` on the `.dvi` file and then print the resulting `.ps` file.
- The `MiKTeX` documentation[4] by Christian Schenk is well written and is available in HTML or PDF formats. There are also a FAQ (frequently asked questions list) and a "tips and tricks" document.
- An archived email discussion list and forum[5] is available on the WWW.

2.2 Other TeX packages for Windows

`emTeX`[6] by Eberhard Mattes is another free TeX implementation for Windows that also supports LaTeX and popular add-ins such as BibTeX. It does not provide PDF support. A freeware Windows

[4] http://www.miktex.org/docs.html
[5] http://sourceforge.net/forum/forum.php?forum_id=33789
[6] ftp://ftp.cdrom.com/pub/tex/ctan/systems/msdos/emtex/README.ENG

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">

shell[7] has been written for this package. I switched to `MiKTeX` from this package because `MiKTeX` appeared more capable and better supported.

`PCTeX`[8] is a commercial TEX package currently selling for $280 that also includes LATEX2e, most of the standard macro packages, and SliTeX fonts. It does not include PDF support. It has no obvious advantages over the free `MiKTeX` package for my purposes so I have never investigated it.

3 LaTeX to HTML conversion

A basic TEX package such as `MiKTeX` supports publication of the document on paper, or on the Web as a passive Postscript or PDF file. It is also increasingly attractive to publish technical documents on the Web with active hyperlinks to other material. There is a role for any package that can convert a single `.tex` master into high-quality printouts, into single-file Postcript or PDF files, or into an easily-navigable HTML file cluster with active hyperlinks, as required.

Documents prepared in LATEX are well suited for automatic conversion into HTML webs. Table 1 illustrates a mapping between document markup tools provided by LATEX and by HTML. Although LATEX offers precise layout control and mathematical markups that cannot yet be used by most web browsers, 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 detailed format) can be mapped uniquely into HTML.

3.1 LaTeX2HTML

LATEX2`HTML` is a collection of `perl` scripts that maps LATEX document structures into a navigable HTML web and also converts content that cannot be represented as HTML into browser-readable

[7] http://www.wfu.edu/Academic-departments/Economics/ftp/emtexgi.html
[8] http://www.pctex.com/prodinfo.htm

images within that web. It was originated by Nikos Drakos[9] at the University of Leeds, U.K. but has since been extended by an *ad hoc* consortium of technical documentation authors and `perl` wizards. It handles the document conversion as follows:

- a separately-installed LaTeX engine (such as that provided by `MiKTeX`) is used to break the document into components as specified by the user's LaTeX sectioning commands, including footnotes, tables of contents, indexing and bibliography;
- the parts of the document that LaTeX2HTML recognizes are converted to the corresponding HTML structures, and are written out, with appropriate hypertext navigation aids, as `.html` files.
- all other parts, including mathematical symbols, equations, graphics and images, are turned into `.dvi` files, `.ps` files and finally into bitmapped `.gif` or `.png` images that are cross-linked to the `.html` as appropriate. (`dvips`, `Ghostscript`, and `netpbm` are invoked during this conversion and therefore must also be installed.)
- 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 used as part of a larger document structure on the WWW or Intranet;
- conditional text structures are provided in an `html.sty` style file, so that the output can be optimized differently for the `.dvi` (and hence PostScript or PDF) and the `.html` versions without further effort by the author.

3.2 Pros and cons of LaTeX2HTML

The features provided with the `html.sty` style allow *one* LaTeX source file to specify a technical document appropriately for publication either

1. as a single `.ps` file, using either LaTeX2e and `dvips`, or
2. as a single `.pdf` file, using `pdflatex`, or
3. as an HTML file cluster, or web, using LaTeX2HTML.

or as an appropriate combination of these.

The document-structuring features of LaTeX provide a very simple way to create, manage and update complex HTML webs with large numbers of internal navigation links, while also ensuring that its contents can be printed in high-quality format from a single file. Authors who are already familiar with LaTeX may therefore find it particularly attractive to use LaTeX2HTML for website management, as it automates the updating of navigation features. (Some large-scale documentation web sites have been built using LaTeX2HTML rather than commercial web management packages, for this very reason).

I have added some NRAO-specific options to LaTeX2HTML that provide for automatic generation of NRAO-standard `.shtml` web pages from LaTeX source, so that the NRAO version of LaTeX2HTML is particularly well suited as a tool for managing longer documents on the NRAO web site.

A minor benefit of using `html.sty` with `pdflatex` is that `html.sty` includes the `hyperref` utility so that the `.pdf` file can provide active links to other documents, whether or not you wish also to create a full-fledged HTML web from it using LaTeX2HTML.

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

Early versions of LaTeX2HTML stumbled on some large documents and the output with all settings defaulted was ugly. The 2K.1beta distribution is more robust and the package is now being developed and documented[10] by an open-source working group led by Ross Moore at McQuarie University in Australia. The documentation is comprehensive (although it lags behind the current version). There is an archived email forum[11] for discussing bugs and development issues, as well as for free (but often well-informed!) help from other users of the package.

With a little care, modern versions of the LaTeX2HTML converter can produce very usable HTML webs from intricate technical documents. A few limitations must be lived with, however:

- The converter assumes that the source uses standard LaTeX2*e* syntax, so legacy documents may need to be worked over before they can be used with LaTeX2HTML. This can become onerous for large documents written with clever, but non-standard, TeX macros;
- The use of inlined images to represent symbols and equations means that the HTML output is neither fully resizable nor fully searchable. The results can look ugly in *some* browser setups, but they are acceptable (and even attractive) under a wide range of circumstances.

3.3 LaTeX2HTML under Windows

All components required to run LaTeX2HTML are now available for Microsoft Windows. In testing here in Charlottesville, the `2K.1beta` version has been shown to work satisfactorily in Windows `NT` and `Win2k` environments with the `MiKTeX`[12] TeX package described in Section 2.1.

To use LaTeX2HTML on a PC, you must first install:

- A TeX implementation: I use `MiKTeX` Version 2.0, released in January 2001. This is available over the net as a free download[13] , and a CD-ROM version is scheduled for release soon. `MiKTeX 2.0` is also archived at `Cvsnap1\SW\texutils\MiKTeX2`.
- A recent Windows implementation of `Ghostscript`[14]. The latest version at time of writing is 6.50; you need a distribution that includes the executable `gswin32c.exe`. `Ghostscript 6.50 for Windows` is archived at `Cvsnap1\SW\texutils\Ghostscript`.
- `Perl`, in a version later than 5.0. I use the freely available ActivePerl[15] which is archived at `Cvsnap1\SW\texutils\ActivePerl`.
- `netpbm` for Windows. This was once supplied with `MiKTeX` as `netpbm.zip` but must now be obtained separately. A copy that provides all of the components needed by LaTeX2HTML is available at `Cvsnap1\SW\texutils\netpbm`. I strongly suggest that you use that version if possible, as there are known compatibility problems with some other variants of `netpbm` for Windows.
- LaTeX2HTML itself, also a free download[16] from the net, or in its NRAO-enhanced variant at `Cvsnap1\SW\texutils\latex2html-2K.1beta`.

To reach the `Cvsnap1` server, click on `Network Neighborhood|Cvsnap1` from a PC in the NRAO NT domain.

A well-written document[17] by Luis Seidel Gómez de Quero and Steve Mayer details how to install LaTeX2HTML alongside `MiKTeX` from external sources (and also mentions some traps for the unwary/unlucky).

[10] http://www-texdev.mpce.mq.edu.au/l2h/docs/manual/
[11] http://www.xray.mpe.mpg.de/mailing-lists/latex2html/
[12] http://www.miktex.org/
[13] http://sourceforge.net/project/showfiles.php?group_id=10783&release_id=20815
[14] http://www.ghostscript.com/
[15] http://www.activestate.com/Products/ActivePerl/download.html
[16] http://saftsack.fs.uni-bayreuth.de/ latex2ht/current/
[17] http://www.mayer.dial.pipex.com/l2h.htm

Appendix A gives detailed installation instructions for each package using the kit on `Cvsnap`.

The installed size of LaTeX2HTML is about 5.5 Mbytes. `Ghostscript` 6.50 is about 18 Mbytes, `netpbm` about 5 Mbytes and *ActivePerl* about 25 Mbytes. The total disk space required for a complete LaTeX2HTML and `MiKTeX` installation is thus about 140 Mbytes.

3.4 Using LaTeX2HTML

With all of these packages installed, you can write a single source file, e.g., `mydoc.tex`, using the `html.sty` style to specify your document, then run whichever combination you need of:

- `latex mydoc` to make a `.dvi` file, followed by `dvips mydoc` to make Postscript output, or
- `pdflatex mydoc` to make PDF output, or
- `latex2html mydoc` to make a self-contained HTML web with built-in navigation aids.

The only minor hiccups known to me in this process are as follows:

- Graphics to be included in the final document must be in the same directory as the source file when running `latex mydoc` but in the target (e.g., `mydoc`) subdirectory when running `latex2html mydoc` with its subdirectory option enabled. This idiosyncracy of LaTeX2HTML under Windows is often mentioned in the email discussion forum, so I presume that an elegant fix for it has eluded even the wizards so far. Two work-arounds are (a) to copy any such graphics files into *both* the input and output directories, or (b) to turn off the program's (default) option to write the output into a new subdirectory! (This problem reportedly does not occur under Linux/Unix.)
- You cannot use all graphics input file formats to produce all output formats. TeX and LaTeX accept Postscript and Encapsulated Postscript graphics, but working with other graphics formats (such as `.jpg`) requires you to insert the Bounding Box information with every graphic. This is awkward at best, so it is better to convert other graphics formats to Postscript before including them in documents to be output as Postscript. Unfortunately, `pdftex` and `pdflatex`, which accept `.jpg, .pdf, .png` or `.tif` graphics, cannot process Postscript graphics.
- URLs that employ characters with special meanings in TeX, such as tilde and underscore, cannot be made fully functional in all output formats. Tilde can be specified in link text as `\~{}` and underscore as `\_`, and will thereafter appear correctly in printed documents and in the HTML that is output by LaTeX2HTML. Unfortunately, however, `pdflatex` remembers the original TeX input syntax and issues it literally as the URL if you click on the link while reading the `.pdf` document in Acrobat.

I am compiling an Authors' Guide[18] for LaTeX2HTML to document methods and tricks that I have found useful when using this package.

I have also tested an NRAO-specific modification to the LaTeX2HTML Perl scripts that provides an option to emit NRAO-standard `.shtml` web pages (using the observatory's official server-side includes). This option lets you convert LaTeX documents directly to the approved NRAO web page format, and can be turned on and off via the LaTeX2HTML initialization file, as described in the Authors' Guide.

3.5 Alternatives to LaTeX2HTML

Several alternatives exist. Some claim to get around the occasional failure of LaTeX2HTML to process clever user-defined TeX macros. No other package appears to have attracted as much world-wide

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

user support as LaTeX2HTML, however. The only one I have looked at in any detail is TeX4HT[19] by Eitan M. Gurar of Ohio State University. TeX4HT also has detailed instructions[20] for co-installation with MiKTeX. It requires installation of the Windows implementation of ImageMagick[21] as well as of the other packages required by LaTeX2HTML. It claims to provide support for XML and MathML.

4 Editing aids

4.1 Windows TeX shell

TexShell32[22] provides a simple but convenient Windows shell for TeX editing and compilation. This is a workable basic interface to the whole TeX and LaTeX package for anyone who's forgotten, or doesn't wish to be reminded, about the MS-DOS Command Prompt. Its editor provides customizable syntax highlighting and templating for TeX and LaTeX structures. The shell also binds TeX compilation, .log file review, and .dvi previewing to one-click buttons.

A notable feature of this shell is that it links the .dvi preview to the position of the cursor in your .tex input file by default, so that you can be, in effect, only a mouse click away from an actual WYSIWYG interface while composing modest-sized TeX or LaTeX documents using this shell.

TexShell32 comes configured for use with MiKTeX and Yap but it could easily be configured for use with other TeX packages by modifying its .ini file. It is written in open-source Delphi by Dirk Struve and Peter Nagel.

4.2 TextPad

With some experience in writing enhanced LaTeX source files using html.sty, HTML webs produced directly by LaTeX2HTML can be used "as is". They will be readable and navigable, even if they are not the most elegant layouts around. It may however sometimes be useful to tweak an HTML web after it has been written out by LaTeX2HTML. While any ASCII text editor can be used to edit single HTML files, I have found that the TextPad editor (routinely installed on NRAO PC's in Charlottesville) has several nice features for LaTeX or LaTeX2HTML users, including:

- it can open and display many files at once (useful when making global changes across a web),
- it can find text or regular expressions in many files at once and produce a clickable list of all instances of the search key (if it had global *replace* as well as this global find, it would truly be an excellent editor for HTML webs),
- its "split" window option lets you view def definitions at the top of the LaTeX source while working simultaneously on the body,
- its line numbering option lets you locate LaTeX source errors quickly from log file error messages,

[19] http://www.tug.org/applications/tex4ht/mn.html
[20] http://facweb.arch.ohio-state.edu/pviton/support/tex4ht.html
[21] http://www.simplesystems.org/ImageMagick/
[22] http://www.projectory.de/texshell/index.html

A NRAO installation kit

Installation packages for all the toolkit components described here are provided on the `Cvsnap1` server in the folder `SW\texutils`. The following sections contain instructions for installing them on a PC where applications reside on the `D:` drive.

Create a `texutils` folder on the `D:` drive. The instructions below assume that you will install all of the packages relevant to LaTeX2HTML under this root. This is not essential, but you will have to modify several of the LaTeX2HTML setup scripts yourself (see Scetion A.6.1) if you do not do this.

A.1 MiKTeX

- Go to `Cvsnap1\SW\texutils\MiKTeX2` and click on `setupwiz.exe` to start the `MiKTeX` Setup Wizard.
- Click on `Next`.
- Tell `MiKTeX` to install into `D:\texutils`, **not** into its default of `D:\Program Files\MiKTeX`. This step is crucial if you will install LaTeX2HTML later, as the LaTeX2HTML installer does not handle the embedded space in the `Program Files` name properly).
- When prompted, tell the Setup Wizard to install `MiKTeX` *shared* (on a multi-user system) to install all components (the default), to create a local `TEXMF` tree on `D:\localtexmf`, and not to incorporate existing trees (unless you are updating an earlier version of MiKTeX).
- Click on `Next` to install the `MikTeX` components.
- At the end of `MiKTeX` component installation, you should see the message `All components have been copied to the installation directory` and the location of an installation report log.
- Tell `MiKTeX` to add its binary directory to the `path`, and click on `Next`.
- Go to `Start|Programs|MikTeX2|DVI Viewer` and register `YAP` as your default `.dvi` viewer.

At this point, you should have a fully functional TeX , LaTeX , `pdftex`, and `pdflatex` environment in which

- Double clicking on any `.dvi` file in a MS `Explorer` window triggers the `Yap` previewer.
- `MiKTeX` and all of its utilities have been installed into subfolders of `D:\texutils`.
- Your path variable has had `D:\texutils\miktex\bin;` prepended to it, so that `tex`, `latex`, `dvips`, `pdftex` and `pdflatex` commands are available while working in any document folder.
- A program group called `MiKTeX2` has been created, with entries for `Help`, for `DVI Viewer` and for `MikTeX Options`.

I suggest that you test this installation by compiling and previewing a test `.tex` source before you install further packages in this suite.

If your path variable has been set correctly, you should be able to open a Command Prompt window and change directory to any folder that contains a `mydoc.tex` source file, then type `tex mydoc`, `latex mydoc`, `pdflatex mydoc`, etc. to produce `.dvi` or `.pdf` output from the `.tex` source. If your installation can not locate the TeX or LaTeX executables in such a test, then your path variable has not been changed appropriately, and later steps in the installation of LaTeX2HTML will fail. The path variable can be changed manually at

`Settings|Control Panel|System|Environment`

`pdflatex` is installed with the output compression level defaulted to `9`, i.e., maximum possible compression. (This can be adjusted if necessary by editing `D:\texutils\pdftex\config\pdftex.cfg` or by inserting `\pdfcompresslevel=N`, with `N` set to `0` through `9`, in the preamble to the document.)

A.2 Windows TEX shell

This utility makes no registry entries and does not use a setup wizard.

Go to `Cvsnap1\SW\texutils\WinTeXShell` and copy its contents to a folder that you create and name `D:\texutils\WinTeXShell`. Then copy the `TexShell.exe` shortcut to wherever you would like to initiate the shell from. (The simplest option is to drag it onto your desktop, but on an NT machine you might want to copy it to `\Winnt\Profiles\AllUsers\Start Menu`.)

At this point, you should have a fully functional Windows-oriented TEX installation. To test it, open the `Windows TeX Shell` and click on `File|Main File`. Select the `.tex` file that you used to test your `MiKTeX` installation, and click `Open`. You should see the source of your file in the editing window. Click on the `TeX` button to compile the file, and on the `Preview` button to view the `.dvi` output with `Yap`.

Proceed to the steps below *only* if you also wish to add the capability for automatic conversion from `.tex` to `.html` with LATEX2HTML.

A.3 Perl

Go to `Cvsnap1\SW\texutils\ActivePerl` and click on `ActivePerl....msi`. This will install `Active Perl` using the Microsoft Installer Version 1.1 or greater. Accept the license and install all features to `D:\Perl`. Tell it to add Perl to your path variable (requires administrator privilege on an NT machine) and to create the Perl file extension association.

If your machine is a Win9x or NT system that does not already have the Microsoft Installer, the installation program will complain immediately, and you should try installing the Installer on your machine. Installers for the Installer are in the `Nt` and `Win9x` subfolders on the installation disk, click on `InstMsi.exe` to activate.

On one NRAO system (`maria` in the Charlottesville Library, running NT4, Service Pack 6), this installation method for `Active Perl 623` failed irretrievably, although the installer backed out of the failure gracefully. `ActivePerl` provides a work-around for cases in which the `.msi` installer fails. To use this, create the folder `D:\Perl` manually, and extract the contents of `ActivePerl....zip` to this folder, preserving the subfolder names from the `. zip`. Then click on `install.bat` in your `D:\Perl` folder. This invokes an installation dialog in a Command Prompt window. Accept the option to modify your path variable to include `D:\Perl`.

A.4 netpbm

Go to `Cvsnap1\SW\texutils\netpbm` and extract all the files in `netpbm.zip` to `D:\texutils`, preserving the subfolder names. This will create `D:\texutils\netpbm` and subfolders populated with various components of the `netpbm` package. No other installation is needed.

Not all `netpbm` packages available for Windows contain all of the features required to run LATEX2HTML, so use of the `netpbm` from `Cvsnap1\SW\texutils\netpbm` is strongly recommended. If you substitute another `netpbm` package for this one, you should monitor the output of the `config` step in LATEX2HTML installation particularly carefully, and watch for any error messages about `netpbm` components. Also look carefully at the output of the first LATEX2HTML run that process images, and make sure that images are displayed and cropped properly.

A.5 Ghostscript

Go to `Cvsnap1\SW\texutils\Ghostscript` and click on `gs650w32.exe` to run the installer for Ghostscript 6.50. Tell it to install Ghostscript in `D:\texutils\gs`, installing all fonts and for all users in an NT environment. Your Ghostscript utilities will then be installed in `D:\texutils\gs\gs6.50` and its subfolders, and fonts in `D:\texutils\gs\fonts`.

A.6 LaTeX2HTML

LaTeX2HTML installation is done in two stages: configuration (and testing), followed by the full installation. The configuration step (Section A.6.1) ensures that LaTeX2HTML scripts can locate all of the parts of the other packages on which it depends. It also provides a simple test of its ability to compile a standard `test.tex` file before you do the full installation (Section A.6.2).

WARNING: LaTeX2HTML *requires components of several packages to co-operate, and you may experience problems if you substitute packages other than those in* `Cvsnap1\SW\texutils`*, or if you name folders differently from the examples in these instructions.*

A.6.1 Configuration and Testing

Go to `Cvsnap1\SW\texutils\latex2html-2K.1beta` and extract all of the contents of the zipped file `latex2html-2K_1nrao.zip`, including sub-folders, to `D:\Temp`. This will create a temporary installation tree under `D:\Temp\l2h`. The LaTeX2HTML installation kit can be run from any folder **on the same disk drive** as `D:\texutils`, provided you avoid folder names that contain embedded spaces. (I suggest using `D:\l2htemp` if you do not want the installer to reside in `D:\Temp` while it is being configured.) If

1. you installed `Ghostscript` in `D:\texutils\gs\gs6.50` and `netpbm` in `D:\texutils\netpbm` as suggested in Sections A.5 and A.4, and
2. you will install LaTeX2HTML to `D:\texutils\l2h`,

then

- open a Command Prompt window,
- change the working folder to `D:\Temp\l2h`, and
- build the installation scripts for LaTeX2HTML by typing `config`.

If you are using other folder names, you must edit the installation preferences file `prefs.pm` to tell LaTeX2HTML where to find these utilities, and where to install itself, before you run `config`. The key edits are on two lines of `prefs.pm`:

- **line 123**
  `$prefs{'EXTRAPATH'}='D:\\texutils\\gs\\gs6.50\\bin;D:\\texutils\\netpbm\\bin';`
  is where you specify the paths to your `Ghostscript` and `netpbm` installations, respectively. The example shows the default locations for the NRAO installation. Note the use of the double backslash when specifying pathnames to be read by a `perl` script.

- **line 129**
  `$prefs{'PREFIX'}='D:\\texutils\\l2h';`
  to specify where to install LaTeX2HTML.

A successful run of `config` produces a log file like this one:

```
Starting Configuration...

config.pl, Release 2K.1beta (Revision 1.33)
Accompanies LaTeX2HTML, (C) 1999 GNU Public License.

checking for old config file (cfgcache.pm)... not found (ok)
```

```
checking for platform... MSWin32 (Windows 32 bit)
checking for D:\Perl\bin\perl.exe... D:\Perl\bin\perl.exe
checking perl version... 5.006
checking if perl supports some dbm... yes
checking if perl globbing works... yes
checking for tex... D:\texutils\miktex\bin\tex.exe
checking for latex... D:\texutils\miktex\bin\latex.exe
checking for initex... D:\texutils\miktex\bin\initex.exe
checking for kpsewhich... no
checking for TeX include path... NONE
Warning: Will not automatically install LaTeX2HTML style files.
checking for dvips... D:\texutils\miktex\bin\dvips.exe
checking dvips version...
5.86
checking if dvips supports the combination of -E and -i -S 1... yes
checking for html4-check... no
checking for gswin32c... \texutils\gs\gs6.50\bin\gswin32c.exe
checking for ghostscript version... 6.50
checking for ghostscript portable bitmap device... pnmraw
checking for full color device for anti-aliasing... ppmraw
checking for ghostscript library and font paths... built-in paths are correct
checking for pnmcrop... \texutils\netpbm\bin\pnmcrop.exe
checking if pnmcrop can crop from one direction... yes
checking for pnmflip... \texutils\netpbm\bin\pnmflip.exe
checking for ppmquant... \texutils\netpbm\bin\ppmquant.exe
checking for pnmfile... \texutils\netpbm\bin\pnmfile.exe
checking for pnmcat... \texutils\netpbm\bin\pnmcat.exe
checking for pbmmake... \texutils\netpbm\bin\pbmmake.exe
checking for ppmtogif... \texutils\netpbm\bin\ppmtogif.exe
checking if ppmtogif can make transparent GIFs... yes
checking if ppmtogif can make interlaced GIFs... yes
checking for pnmtopng... \texutils\netpbm\bin\pnmtopng.exe
checking if multiple pipes work... no
Unfortunately multiple pipes are not reliable on this OS.
checking for temporary disk space... C:\TEMP
creating cfgcache.pm
creating test.bat
creating install.bat
Note: Will install...
      ... executables to   : D:\texutils\l2h\bin
      ... library items to : D:\texutils\l2h
Starting build...
... building latex2html
build.pl (Revision 1.5)
Building "latex2html.bat" from "latex2html.pin"
... building pstoimg
build.pl (Revision 1.5)
Building "pstoimg.bat" from "pstoimg.pin"
... building texexpand
build.pl (Revision 1.5)
Building "texexpand.bat" from "texexpand.pin"
... building configuration module
```

```
build.pl (Revision 1.5)
Building "l2hconf.pm" from "l2hconf.pin"
Configuration procedure finished
```

Check this log carefully. This is where you confirm that LaTeX2HTML can locate everything it needs for a general LaTeX to HTML translation, including the display of non-HTML symbols and graphics as embedded images.

In the above example, `config` found, TeX, LaTeX, and `initex` in the folder specified by the current `MiKTeX` search path. If any of those steps give an error message, LaTeX2HTML will not be able to process any `.tex` file.

In the example, `config` also located the `dvips`, `gswin32c`, and `netbpm` utilities that are required for proper image manipulation. Any error messages about these utilities will prevent LaTeX2HTML from processing embedded images correctly. Note that `config` currently has a quirk that makes it pause on most Windows machines while it is searching for `dvips`; hitting `ENTER` two or three times when it pauses should get `config` going again.

A successful `config` run creates several `perl` scripts customized for your particular installation, as batch files.

If `config` produces any error messages saying that a package is not found, you *must* go back and edit line 123 of `prefs.pm` and/or your `MiKTeX` path setup accordingly.

If `config` completes with no error messages, then execute `test.bat` to feed LaTeX2HTML a built-in `test.tex` file. Use your web browser to examine the `test.html` that this produces. This file tests the rendering of several fonts, of a table structure, and of some simple in-line images. Don't worry about the images in this test having gray backgrounds while the rest of the document is white. This will be corrected in the real installation that follows.

A.6.2 Installation

If `config` and `test` have run successfully, run `install.bat` from `D:\Temp\l2h` to install a full copy of LaTeX2HTML under the folder specified by `$prefs{'PREFIX'}` on line 129 of `prefs.pm`. When the installation finishes, you should see the message:

```
Done.  Have a lot of fun with LaTeX2HTML!
```

You can now delete the temporary installation structure that was built in `D:\Temp\l2h` if you want to save disk space, but if you edited any of the installation scripts I recommend keeping it around for a while until you are sure that your LaTeX2HTML does everything that you want with real documents.

You should also copy the new `.sty` files installed by LaTeX2HTML in `D:\texutils\l2h\texinputs` into your `D:\localtexmf\tex\l2h` folder. To allow `MiKTeX` to access them when running `latex` and `pdflatex`, do the following:

- Click on `Start|Programs|MiKTeX2|MikTeX Option`,
- Go to the `Roots` tab.
- Click on a non-root entry such as `D:\texutils` and then the `Add` button.
- Select `D:\localtexmf\tex\l2h` from the folder tree window, and it will be added to the folder list to be searched by `MiKTeX` when it runs.
- Click on `Apply`,
- Return to the `General` tab and click `Refresh Now` to refresh the database.

Finally, add `D:\texutils\l2h\bin;` to your path to make the `latex2html` command available from any document folder (this needs administrator privilege on an NT machine).

You should now be able to run `latex`, `pdflatex` or `latex2html` on a `.tex` input file to produce the appropriate output format/ This is a good time to test this by running all of them on a file tht you know to be syntactically correct..

In practice, you may wish to customize LaTeX2HTML operation to your particular preferences, or for a particular document. The package offers you a wide range of options for customization, through the initialization file(s), as discussed in Section A.6.3

A.6.3 Initialization

File name

The file `l2hconf.pm` in the LaTeX2HTML root directory can be used to set switches that give you fine control over how documents are processed by LaTeX2HTML. If you installed from a kit other than the NRAO-specific kit on `Cvsnap1`, you will probably find it convenient to edit line 187 to read:

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

so that the program will look for an initialization file called `l2h.ini` in the document directory to control program settings. The default for `INIT_FILE_NAME` is the Unix-style `.latex2html-init` which does not parse well on Windows systems.

Image format

Another default worth changing in `l2hconf.pm` is to edit line 52 to read:

```
@IMAGE_TYPES = qw(gif png);
```

which makes LaTeX2HTML use transparent `.gif` files for its self-generated images. The default produces `.png` files that look ugly against anything but the default gray background.

B NRAO customization of LaTeX2HTML

I have made several minor modifications to the standard LaTeX2HTML distribution for use and distribution at the NRAO.

B.1 Option to use l2h.ini as the initialization file

The NRAO distribution is preconfigured to expect `l2h.ini` as its initialization file and comes with a `l2h.ini` that is identical to the `.latex2html-init` file in the original distribution.

The changes made to the standard distribution kit to do this were:

- Set `$INIT_FILE_NAME = 'l2h.ini';` in `l2hconf.pin`, line 196
- add the line `'l2h.ini' => 'lib',` at line 204 of `install.pl`, and
- provide a duplicate copy of `.latex2html-init` in the NRAO installation kit as `l2h.ini`.

B.2 Option to emit NRAO-standard web pages

When using LaTeX2HTML to publish documents 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. I therefore modified the LaTeX2HTML installation input script `latex2html.pin` for use at the NRAO as follows:

Just before the standard line

```
join("\n", (($DOCTYPE)? $DTDcomment : '' )
```

at line 7013, add the following:

```
#   insert NRAO style and banners if requested
    if ($NRAO) {$DTDcomment = "<!--#include virtual=\"/nrao-header.html\"-->"};
    if ($NRAO) {$begin_head = "-->\n<TITLE>"}
    else {$begin_head = " -->\n<HTML>\n<HEAD>\n<TITLE>"};
    if ($NRAO) {$end_head = "<!--#include virtual=\"/nrao-banner.html\"-->"}
    else {$end_head = "</HEAD>"};
    if ($NRAO) {$begin_foot = "<!--#include virtual=\"/nrao-trailer.shtml\"-->\n"}
    else {$begin_foot=""};
    if ($NRAO) {$end_foot = "<!--#include virtual=\"/nrao-end.html\" -->\n"}
    else {$end_foot = "\n</BODY>\n</HTML>\n"};
```

and modify the `join` segment (at line 7024 in the revised file) to read

```
    join("\n", (($DOCTYPE)? $DTDcomment : '' )
,"<!--Converted with LaTeX2HTML $TEX2HTMLVERSION"
, "original version by:  Nikos Drakos, CBLU, University of Leeds"
, "* revised and updated by:  Marcus Hennecke, Ross Moore, Herb Swan"
, "* with significant contributions from:"
, "  Jens Lippmann, Marek Rouchal, Martin Wilck and others"
, "  including NRAO-specific modifications by Alan Bridle"
$begin_head.$title."</TITLE>"                # NRAO mod
, &meta_information($title)
,  ($CHARSET && $HTML_VERSION ge "2.1" ?
"<META HTTP-EQUIV=\"Content-Type\" CONTENT=\"text/html; charset=$this_charset\">"
 : "" )
, $LATEX2HTML_META
, ($BASE ? "<BASE HREF=\"$BASE\">" : "" )
, $STYLESHEET_CASCADE
, ($STYLESHEET ? "<LINK REL=\"STYLESHEET\" HREF=\"$STYLESHEET\">" : '' )
, $more_links_mark
```

```
, $end_head , ($before_body? $before_body : '')   # NRAO mod
, "<BODY $body>", '');
```

Modify the make_real_address routine at line 746 in the revised file to read

```
sub make_real_address {
    local($addr) = $ADDRESS;
    if ((defined &custom_address)&&($addr)) {
&custom_address($addr)
    } elsif ($addr) {
$begin_foot."<ADDRESS>\n$addr\n</ADDRESS>";   # NRAO mod
    } else { '' }
}
```

Modify the make_address routine at line 7139 in the revised file to read

```
sub make_address {
    local($addr) = &make_real_address(@_);
    $addr .= $end_foot;                          # NRAO mod
    &lowercase_tags($addr) if $LOWER_CASE_TAGS;
    $addr;
}
```

Modify the make_file routine at line 8442 in the revised file to read

```
sub make_file {
    # Uses and modifies $_ defined in the caller
    local($filename, $title, $layout) = @_;
    $layout = $BODYTEXT unless $layout;
    $_ = join('',&make_head_and_body($title,$layout), $_
, (($filename =~ /^\Q$footfile\E$/) ? '' : &make_address )
, (($filename =~ /^\Q$footfile\E$/) ? $end_foot : '')        # NRAO mod
);
    &replace_markers unless ($filename eq $footfile);

    unless(open(FILE,">$filename")) {
        print "\nError: Cannot write '$filename': $!\n";
        return;
    }
    print FILE $_;
    close(FILE);
}
```

Note that if you did not get your copy of LaTeX2HTML from the NRAO toolkit, you can add this NRAO option to your local copy after it has been installed, by directly editing these changes into your own `latex2html.bat`.

These NRAO-specific options can then be turned on by setting the variables

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

in the initialization file. The first activates the added options, and the second tells LaTeX2HTML to write out 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 revert to the default (i.e., non-NRAO-specific) LaTeX2HTML output.

Index