The LaTeX2HTML Translator

Nikos Drakos,
Computer Based Learning Unit,
University of Leeds.

January 26, 1994

This document accompanies **LaTeX**2HTML Version 0.5.3.

Abstract

LaTeX2HTML is a conversion tool that allows documents written in LaTeX to become part of the WorldWide Web. In addition, it offers an easy migration path towards authoring complex hypermedia documents using familiar word-processing concepts.

LaTeX2HTML replicates the basic structure of a LaTeX document as a set of interconnected HTML files which can be explored using automatically generated navigation panels. The cross-references, citations, footnotes, the table of contents and the lists of figures and tables, are also translated into hypertext links. Formatting information which has equivalent "tags" in HTML (lists, quotes, paragraph breaks, type styles, etc.) is also converted appropriately. The remaining heavily formatted items such as mathematical equations, pictures or tables are converted to images which are placed automatically at the correct positions in the final HTML document.

LaTeX2HTML extends LaTeX by supporting arbitrary hypertext links and symbolic cross-references between evolving remote documents. It also allows the specification of *conditional text* and the inclusion of raw HTML commands. These hypermedia extensions to LaTeX are available as new commands and environments from with a LaTeX document.

This document presents the main features of **LaTeX**2HTML , and describes how to obtain, install, and use it.

Contents

1 Overview 4

2 User Manual 5
2.1 Command Line Options . . . 5
2.2 Extending the Translator . . . 7
2.2.1 Adding Support for Specific Style Files . . . 7
2.2.2 Asking the Translator to Ignore Commands . . . 8
2.2.3 Asking the Translator to Pass Commands to LaTeX . . . 8

3 Special Features 9
3.1 Hyperlinks in LaTeX . . . 9
3.2 Cross-References Between Living Documents . . . 9
3.2.1 Example . . . 10
3.3 Including Arbitrary HTML Markup . . . 11
3.4 Conditional Text . . . 12
3.5 Cross References Shown as “Hyperized” Text . . . 13
3.6 Customizing the Navigation Panel . . . 14

4 Getting LaTeX2HTML 15

5 Requirements 16

6 Installing LaTeX2HTML 17

7 Changes from Previous Versions 18
7.1 Changes from 0.5.1 to 0.5.3 . . . 18
7.2 Changes from v0.4 to v0.5.1 . . . 19
7.3 Changes from v0.3.1 to v0.4 . . . 19
7.4 Changes from v0.3 to v0.3.1 . . . 21
7.5 Changes from v0.2 to v0.3 . . . 22
7.6 Changes from v0.1.1 to v0.2 . . . 24
7.7 Changes from v0.1 to v0.1.1 . . . 25

8 Known Problems 25

9 Troubleshooting 26

10 Sample Converted Documents 30

11 Figures, Tables and Arbitrary Environments 32

12 General License Agreement and Lack of Warranty 32

13 Credits 33

List of Figures

1 A sample figure 35

List of Tables

1 A sample table 32

1 Overview

This document describes a LaTeX to HTML translator written in Perl.

The translator:

- breaks up a document into one or more components as specified by the user[1],
- provides optional iconic navigation panels on every page which contain links to other parts of the document,
- handles inlined equations ($\sum_{i=1}^{n} x_i = \int_0^1 f$), right-justified numbered equations (see equation 2), tables (see Table 1), or figures (see Figure 1), and *any arbitrary environment*[2],
- can produce output suitable for browsers that support inlined images or character based browsers (as specified by the user),
- handles definitions of new commands, environments, and theorems even when these are defined in external style files[3],
- handles footnotes[4], tables of contents, lists of figures and tables, bibliographies, and can generate an index,
- translates cross-references into hyperlinks and extends the LaTeX cross-referencing mechanism to work not just within a document but *between documents* which may reside in remote locations,
- translates LaTeX accent and special character commands (e.g. Ȧ Øö £©¶) to the equivalent ISO-LATIN-1 character set where possible,
- recognizes hypertext links (to multimedia resources or arbitrary internet services such as sound/video/ftp/http/news) and links which invoke arbitrary program scripts, all expressed as LaTeX commands,
- recognizes *conditional text* which is intended only for the hypertext version, or only for the paper (DVI) version,
- can include raw HTML in a LaTeX document (e.g. in order to specify interactive forms),
- can deal sensibly at least with the *Common* LaTeX *commands* summarized at the back of the LaTeX blue book [1],

[1] The user can specify the *depth* at which the document should not be broken up any further.

[2] These are passed to LaTeX and then converted to images which are either included in the document or are made available through hypertext links.

[3] This allows the definition of HTML macros in LaTeX !

[4] Like this!

- will try and translate any document with embedded LaTeX commands irrespective of whether it is complete or syntactically legal.

2 User Manual

To use **LaTeX2HTML** simply type `latex2html <file>.tex`. This will create a new directory called `file` which will contain the generated HTML files, some log files and possibly some images. To view the result use an HTML viewer such as NCSA Mosaic[5] on the main HTML file which is `file/file.html`. This file will contain navigation links to the other parts of the generated document.

It is possible to customize the output from **LaTeX2HTML** using a number of command line options (see Section 2.1) with which you can specify how to break up the document, where to put the generated files, what the title is, what the signature at the end of each page is, what to put in the navigation panel, what kind of extra information to include about the document, whether to retain the original (LaTeX) section numbering scheme, etc.

Also, the **LaTeX2HTML** script includes a short manual which can be viewed by saying `%nroff -man latex2html` .

2.1 Command Line Options

The command line options described below can be used to change the default behavior of **LaTeX2HTML** . Alternatively, any corresponding environment variables (see Section6) in the initialization file `.latex2html-init` may be changed, in order to achieve the same result.

-split num Stop splitting sections into separate files at this depth. A value of 0 will put the document into a single HTML file. The default is 8.

-link num Stop revealing child nodes at each node at this depth. (A node is a part/chapter/section/subsection/subsubsection etc.). A value of 0 will show no links to child nodes, a value of 1 will show only the immediate descendents, etc. A value at least as big as that of the `-split` option will produce a table of contents for the tree structure, rooted at each given node. The default is 4.

-nolatex Disable the mechanism for passing unknown environments to LaTeX for processing. This can be thought of as "draft mode" which allows faster translation of the basic document structure and text, without fancy figures, equations or tables.

-external_images Instead of including any generated images inside the document, leave them outside the document and provide hypertext links to them.

[5] http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/help-about.html

-ascii_mode Use only ascii characters and do not include any images in the final output. In ascii mode the output of the translator can be used on character based browsers which do not support inlined images (the <IMG> tag).

-allbitmaps Generate images for figures, tables, and tabular environments as XBMs rather than GIFs (the default). This may increase the disk space required to store the images but will allow them to blend naturally into a colored background.

-t top-page-title Name the document using this title.

-dir output-directory Redirect the output to this directory.

-address author-address Sign each page with this address.

-no_navigation Disable the mechanism for putting navigation links in each page.

-top_navigation Put navigation links at the top of each page.

-bottom_navigation Put navigation links at the bottom of each page AS WELL as the top.

-auto_navigation Put navigation links at the top of each page. If the page exceeds `$WORDS_IN_PAGE` number of words (the default is 450) then put one at the bottom of the page as well.

-index_in_navigation Put a link to the index page in the navigation panel if there is an index.

-contents_in_navigation Put a link to the table of contents in the navigation panel if there is a table of contents.

-next_page_in_navigation Put a link to the next logical page in the navigation panel.

-previous_page_in_navigation Put a link to the previous logical page in the navigation panel.

-info string Generate a new section *About this document ...* containing information about the document being translated. The default is to generate such a section with information on the original document, the date, the user and the translator. An empty string (or the value 0) disables the creation of this extra section. If a non-empty string is given, it will be placed in the contents of the *About this document ...* section instead of the default information.

-dont_include file(s) Do not include the specified file(s). Such files are usually style files which may contain raw TEX commands that LATEX or the translator cannot handle.

-reuse Reuse images generated during previous translations where appropriate. This also disables the initial interactive session during which the user is asked whether to reuse the old directory, delete its contents or quit. Images which depend on context (e.g. numbered tables or equations) cannot be reused and are always re-generated.

-init_file file Load the specified file. This Perl file will be loaded after loading $HOME/.latex2html-init (if one exists). It can be used to change the default options.

-show_section_numbers Show section numbers. By default the section numbers are not shown in order to encourage the use of particular sections as standalone documents. In order to be shown, section titles must be unique and must not contain inlined graphics.

-h Print out the list of options.

2.2 Extending the Translator

As the translator covers only partially the set of LATEX commands and because new LATEX commands can be defined arbitrarily using low level TEX commands, the translator should be flexible enough to allow end users to specify how they want particular commands to be translated.

2.2.1 Adding Support for Specific Style Files

LaTeX`2HTML` provides a mechanism where code to translate specific style files is automatically loaded if such code is available. When the use of a style file such as `german.sty` is detected in a LATEX source document, the translator looks for a file `LATEX2HTMLDIR/styles/german.perl`. If one is found, then it will be loaded into the main script.

This mechanism will help to keep the core script smaller as well as make it easier for others to contribute and share solutions on how to translate specific style files. The current distribution includes the files `german.perl`, `french.perl`, `html.perl` and `makeidx.perl`.

The problem however, is that writing such extensions requires an understanding of Perl and of the way **LaTeX**`2HTML` is organized. More user-friendly interfaces will be investigated.

At the moment a rudimentary mechanism is provided so that a user can ask for particular commands and their arguments either to be ignored or passed on to LATEX for processing (the default behavior for unrecognized commands is for their arguments to remain in the HTML text). Commands that are passed on to LaTeX are converted to images which are either "inlined" in the main document or become accessible via hypertext links. Simple extensions using the commands above may be included in the `LATEX2HTMLDIR/latextohtml.config` file or in each personal `HOME/.latex2html-init` initialization file.

2.2.2 Asking the Translator to Ignore Commands

Commands that should be ignored may be specified in the `.latex2html-init` file as input to the `ignore_commands` subroutine. Each command which is to be ignored should be on a separate line followed by compulsory or optional argument markers separated by `#`'s e.g.[6]:

```
<cmd_name>#{}# []# {}# [] ...
```

`{}`'s mark compulsory arguments and `[]`'s optional ones.

Some commands may have arguments which should be left as text even though the command should be ignored (e.g. `mbox`, `center`, etc). In these cases the arguments should be left unspecified.

Here is an example of how this mechanism may be used:

```
&ignore_commands( <<_IGNORED_CMDS_);
documentstyle # [] # {}
linebreak# []
center
<add your commands here>
_IGNORED_CMDS_
```

2.2.3 Asking the Translator to Pass Commands to LaTeX

Commands that should be passed on to LaTeX for processing because there is no direct translation to HTML may be specified in the `.latex2html-init` file as input to the `process_commands_in_tex` subroutine. The format is the same as that for specifying commands to be ignored. Here is an example:

```
&process_commands_in_tex (<<_RAW_ARG_CMDS_);
fbox # {}
framebox # [] # [] # {}
<add your commands here>
_RAW_ARG_CMDS_
```

[6] It is possible to add arbitrary Perl code between any of the argument markers which will be executed when the command is processed. For this however a basic understanding of how the translator works and of course Perl is required.

3 Special Features

This section describes some special features of the translator which can be considered as *hypertext extensions to* LaTeX. These are implemented as new LaTeX commands which have special meaning during the translation to HTML but are mostly ignored when processed by LaTeX.

All the new commands used in the sections below are defined in the file `html.sty`[7] which is part of the distribution. *This file must be included in any* LaTeX *document using these features*[8].

3.1 Hyperlinks in LaTeX

Arbitrary hypertext references are available through the commands `htmladdnormallink` and `htmladdimg`).

These are defined as follows:

```
\htmladdnormallink{<text>}{<URL>}
\htmladdimg{<URL>}
```

The first command expects some text as the first argument and a URL as the second argument. When processed by LaTeX (i.e. in the `dvi` or `ps` output files), the URL will appear as a footnote to the text. But when processed by the translator, the URL will be used to provide an active hypertext link[9] (to another file, picture, sound file, movie, etc) e.g.

```
\htmladdnormallink{<URL>}
{http://www.ncsa.uiuc.edu/demoweb/url-primer.html}
```

In a similar way, the argument of the `htmladdimg` command should be a URL pointing to an image. This URL is ignored in the LaTeX hard copy output.

3.2 Cross-References Between Living Documents

This section describes a mechanism for establishing cross-references between two or more documents via symbolic labels which are independent of the physical realisation of these documents. The documents involved may reside in remote locations and may be spread across one or more HTML files.

The mechanism is an extension of the simple `label-ref` pairs which can be used only within a single document. A symbolic label defined with a `label` command can be referred to using a `ref` command. When processed by LaTeX each `ref` command is replaced by the section number in which the corresponding `label` occurred. When processed by the translator, each `ref` is replaced by a

[7] http://cbl.leeds.ac.uk/nikos/tex2html/doc/html.sty.txt

[8] As an argument to the `documentstyle` command or as an argument to the `input` command or by pasting the definitions of individual commands directly into the text!

[9] http://www.ncsa.uiuc.edu/demoweb/url-primer.html

hypertext link to the place where the corresponding `label` occurred. The new commands detailed below show how it is possible to refer to symbolic labels defined with `label` in other external documents. Such references to external symbolic labels are then translated into hyperlinks pointing to the external document.

Cross-references between documents are established with the commands:

```
\externallabels{<URL to directory of external document>}
               {<external document labels.pl file>}

\externalref{<label in remote document>}
```

The first argument to `externallabels` should be a URL to the directory containing the external document. The second argument should be the full pathname to the `labels.pl` file belonging to the external document. The file `labels.pl` associates symbolic labels with physical files and is generated automatically for each translated document[10]. The command `externallabels` can be used once for each external document in order to import the external labels into the current document. The argument to `externalref` may be any symbolic label defined in any of the external documents in the same way that the `ref` command refers to labels defined internally.

After any modifications in an external document (sections added/deleted, segmentation into different physical parts etc.) a new `labels.pl` will be generated. If the `externallabels` command in another document contains the correct address to the `labels.pl` file then the cross-references will be realigned after running the local document through the translator. A warning is given if `labels.pl` cannot be found.

3.2.1 Example

To understand this mechanism better consider how you would maintain a link to this section (of the hypertext version of this document) from one of your documents without using labels. Sure enough you can get the name of the physical file that this section is in. This however is very likely to change and any links to it will become invalid. To update your link, the name of the new file must be found and your link changed by hand. Also there is no general updating mechanism so the only way to find out if your document is pointing to the right place is by actually following the link and then doing a manual update[11].

Now consider how it could be done with symbolic labels. First you have to import the labels used in this document by copying the file labels.pl[12], saving

[10] Note that for *remote* external documents it is necessary to copy the `labels.pl` file locally so that it can be read when processing a local document that uses it.

[11] Link validation can be done automatically but the updating must be done manually when filenames have changed (assuming no other symbolic label mechanism is available).

[12] http://cbl.leeds.ac.uk/nikos/tex2html/doc/latex2html/labels.pl

it say in /tmp/labels.pl, and then adding anywhere in your document:

```
\externallabels{http://cbl.leeds.ac.uk/nikos/tex2html/doc/latex2html}
               {/tmp/labels.pl}
```

After that you can use the label `crossrefs` defined at the beginning of this section[13] as follows:

```
\externalref{crossref}
```

This will be translated into the appropriate hyperlink to this page. If there are any changes in this document and you would like to bring your document upto date you have to copy labels.pl[14] again and rerun the translator on your document. Of course if I move the directory containing the HTML files for this document somewhere else then you would have to make a change in the argument of the `externallabels` command to reflect this.

It is obvious that some level of collaboration is required between authors trying to maintain cross-references between different documents. Using symbolic labels makes this a lot easier (especially for documents written by the same author).

3.3 Including Arbitrary HTML Markup

Raw HTML or HTML+ tags and text may be included using the environment `rawhtml`. This can be used to take advantage of new HTML+ facilities as soon as they become available. A particularly good use of this feature is in the creation of interactive electronic forms[15] from within a LaTeX document. When producing the paper (DVI) version of a document the `rawhtml` environment is ignored.

Here is an example:

```
\begin{rawhtml}
<HR>
<FORM ACTION="http://cbl.leeds.ac.uk/nikos/doc/error.html">
<OL>
<LI> <INPUT TYPE="checkbox" NAME="wp" VALUE="word"> Word for
Windows.
<LI> <INPUT TYPE="checkbox" NAME="wp" VALUE="wp"> Word Perfect.
<LI> <INPUT TYPE="checkbox" NAME="wp" VALUE="latex"> LaTeX.
<LI> Plain Text Editors (Please Specify): <INPUT TYPE="text" NAME="other_ed">
</OL>
```

[13] You either have to guess the role of each label by looking at the `labels.pl` file or by asking the author!

[14] http://cbl.leeds.ac.uk/nikos/tex2html/doc/latex2html/labels.pl

[15] http://south.ncsa.uiuc.edu/forms.html

```
So, what do think (comments please): <BR>
<INPUT TYPE="text" SIZE=45,4 NAME="other_wp">

<INPUT TYPE="submit" VALUE="submit this form but don't expect much!">
</FORM>
<HR>
\end{rawhtml}
```

The result is shown below:

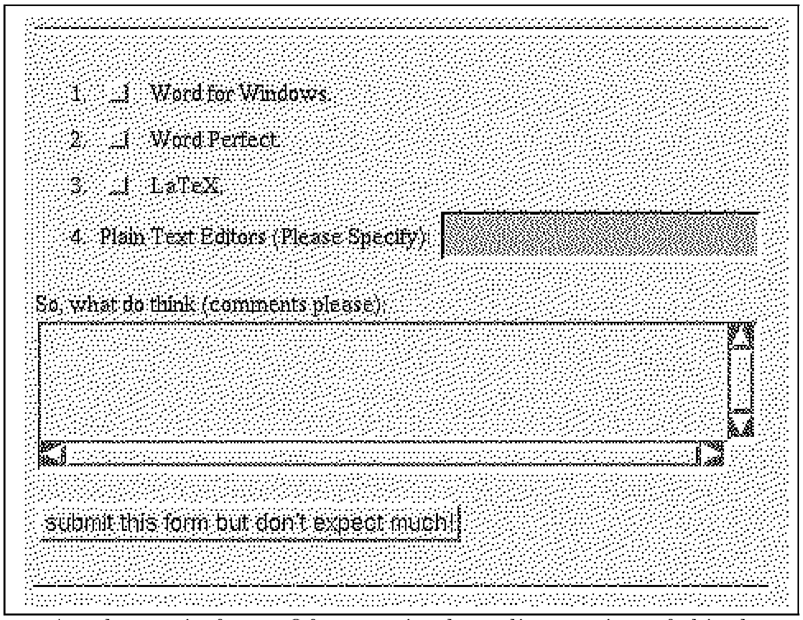

Figure: An electronic form. Of course in the online version of this document the form above would be active.

Warning: Avoid using LaTeX commands involving counters (e.g. numbered figures or equations) in conditional text because this may disrupt the values of the counters in the electronic version.

3.4 Conditional Text

Conditional text can be specified using the environments `latexonly` and `htmlonly`. These allow writing parts of a document which are intended only for electronic delivery or only for paper-based delivery.

This would be useful for example in adding a long description of a multimedia resource in the paper version of a document. Such a description would be redundant in the electronic version, as the user can have direct access to this resource.

Here is an example of the use of the `latexonly` environment:

```
\begin{latexonly}
\begin{figure}
\psfig{file=eform.ps,width=4in}
\caption{An electronic form. Of course in the online version
of this document the form above would be active!}
\end{figure}
\end{latexonly}
```

3.5 Cross References Shown as "Hyperized" Text

In printed documents cross-references are shown through a *numerical or symbolic indirection* e.g. "see Figure 1" (numeric indirection), or "see section "Changes" (symbolic indirection). **LaTeX**2HTML can mirror this mechanism using the same numeric or symbolic references, or when these are not appropriate by using iconic references.

In a hypertext document however, cross-references can be shown without any indirection just by highlighting a relevant piece of text. This can make a document more readable as it removes unnecessary information.

A single new LaTeX command **hyperref** can be used for specifying how a cross-reference should appear both in the printed document and in the hypertext version.

Assuming that the label `sec:cond` is defined somewhere in a document the command **hyperref** which takes 4 arguments can be used in the same document as follows:

```
{\em Is the concept of
\hyperref
% This will be highlighted in the hypertext version
{conditional text} % argument #1
% This will be shown in the printed version
% followed by a numeric reference ...
{conditional text (see Section }   % argument #2
% ... followed by this text
{ for more information)}  % argument #3
% This the common label
{sec:cond} % argument #4
a good idea? }
```

Here is how it will be shown:
Is the concept of conditional text (see Section 7.3 for more information) a good idea?

In the hypertext version what would appear is:
Is the concept of **conditional text** *a good idea?*
Of course **conditional text** would be an active hypertext link.

3.6 Customizing the Navigation Panel

The navigation panel is the strip containing “buttons” and text that appears at the top and perhaps at the bottom of each generated page and provides hypertext links to other sections of a document. Some of the options and variables that control whether and where it should appear have already been mentioned in Section 2.1.

Apart from these general facilities it is also possible to specify completely what appears in the navigation panel and in what order. As each section is processed, **LaTeX**2HTML assigns relevant information to a number of global variables. These variables are used by the subroutine `navigation_panel` where the navigation panel is constructed as a string consisting of these variables and some formatting information.

This subroutine can be redefined in a system or a user configuration file (`LATEX2HTMLDIR/latex2html.config` and `HOME/.latex2html-init` respectively). *Any combination of text, HTML tags, and the variables mentioned below is acceptable.*

The control panel variables are:
Iconic links (buttons)

- 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

Textual links (section titles)

- PREVIOUS_TITLE - points to the previous section
- UP_TITLE - points up to the ”parent” section
- NEXT_TITLE - points to the next section
- NEXT_GROUP_TITLE - points to the next ”group” section
- PREVIOUS_GROUP_TITLE - points to the previous ”group” section

If the corresponding section exists each iconic button will contain an active link to that section. If the corresponding section does not exist, the button will be inactive. If the section corresponding to a textual link does not exist then the link will be empty.

The number of words that appears in each textual link is controlled by the variable `WORDS_IN_NAVIGATION_PANEL_TITLES` which may also be changed in the configuration files.

Below is an example of a navigation panel (the "." is the Perl string concatenation operator and "#" signifies a comment).

```
sub navigation_panel {

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

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

    "<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) .

    #  Horizontal rule (3-d dividing line) and new paragraph
    "<HR> <P>\n"
}
```

4 Getting LaTeX2HTML

First source code[16] and save it in a file (e.g. `latex2html.tar`)[17].

Then, extract the files with

```
% tar xvf latex2html.tar
```

You should then have the following:

[16] http://cbl.leeds.ac.uk/nikos/tex2html/latex2html.tar

[17] The source code is also available via anonymous ftp[18].

- A `README` file.
- A `Changes` file.
- The `latex2html` Perl program.
- A `texexpand` Perl program [19].
- A `latex2html.config` configuration file.
- An `install-test` installation and testing Perl script.
- A `dot.latex2html-init` sample initialization file.
- A `html.sty` style file.
- A `pstogif` shell script.
- A `pstoxbm` shell script.
- A `pstoppm.ps` file
- A `doc/latex2html.ps` postscript file.
- An `icons/` subdirectory containing various icons.
- A `styles/` subdirectory containing Perl code for handling some style files.

5 Requirements

The translator makes use of several programs all of which are freely available on most platforms. The requirements for using **LaTeX**`2HTML` depend on the kind of translation it is asked to perform as follows:

1. **LaTeX commands but without equations, figures, tables, etc.**

   - `Perl`[20] (version 4.0 - RCSfile: perl.c,v - Revision: 4.0.1.8 - Date: 1993/02/05 19:39:30 - Patch level: 36)

     Warning: You really DO need Perl at patch level 36 (see Section 9)

2. **LaTeX commands with equations, figures, tables, etc.**
   As avove plus

   - `latex`,

[19] Written by Robert S. Thau <rst@edu.mit.ai>. The original version of `texexpand` is part of any TeX distribution.
[20] ftp://ftp.uu.net/languages/perl/

- `dvips`[21] (version 5.516 or later),
  Warning: You really DO need `dvips` version 5.516 (see Section 9)
- `gs` (Ghostscript version 2.6.1).
- The `pbmplus`[22] library (distribution of 22nov89 or later). From this, the utilities `pnmcrop`, `pbmtoxbm`, `ppmtogif` are used during the postscript to XBM/GIF conversion.

3. **Right justified equations**
   As above plus the following `pbmplus` utilities: `xbmtopbm`, `pnmfile`, `pnmtile`, and `pnmcat`.

Because by default the translator makes use of inlined images in the final HTML output, it would be better to have a viewer which supports the <`IMG`> tag, such as NCSA Mosaic[23]. If only a character based browser is available or if you want the generated documents to be more portable then the translator can be used with the `-ascii_mode option (See Page`2.1).

If `ghostscript` or the `pbmplus` library are not available it is still possible to use the translator with the `-nolatex option (See Page`2.1).

If you intend to use any of the special features of the translator (see Page 9) then you have to include the `html.sty` file in any LaTeX documents that use them.

6 Installing LaTeX2HTML

To install **LaTeX**`2HTML` you **MUST** do the following:

1. **Specify where Perl is on your system.**
   In each of the files `latex2html`, `texexpand` and `install-test`, modify the second line according to where Perl is on your system. If you have Perl installed as a shell you can replace the first three lines in each file with `#!<perl-directory>/perl`.

2. **Specify where the external programs are on your system.**
   In the file `latex2html.config` give the correct pathnames for some directories (the `latex2html` directory and the `pbmplus` library) and some executables (`latex, dvips, gs`). Note that it is possible to use **LaTeX**`2HTML` even if you don't have some of the external programs. While you're at it you may want to change some of the default options in the same file.

[21] ftp://ftp.tex.ac.uk/pub/archive/dviware/dvips
[22] ftp://ftp.x.org/contrib/
[23] http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/help-about.html

3. **Run `install-test`.**
   This Perl script will make some changes in the `latex2html` file and then check whether the pathnames that should be specified during the previous step are correct.

This is enough for the main installation but you may also want to do some of the following:

- **To use the new LaTeX commands in `html.sty`:**
  Make sure that LaTeX knows where the html.sty file is, either by putting it in the same place as the other style files on your system, or by changing your TEXINPUTS shell environment variable.

- **To set up a different initialization file for each user:**
  Copy the file `dot.latex2html-init` in the home directory of any user that wants it, modify it according to her preferences and rename it as `.latex2html-init`. At runtime, both the `latex2html.config` file and `$HOME/.latex2html-init` file will be loaded, but the latter will take precedence.

- **To make your own local copies of the LaTeX2`HTML` icons:**
  Please copy the `icons` subdirectory to a place under your WWW tree where they can be served by your server. Then modify the value of the `$ICONSERVER` variable in `latex2html.config` accordingly.

  Warning: If you cannot do that bear in mind that these icons will have to travel from Leeds !!! Also, your documents will depend on our server being operational in the first place.

7 Changes from Previous Versions

The previous versions of the translator and some patches[24] are available. A detailed list of changes[25] is also available with the **LaTeX2**`HTML` distribution.

7.1 Changes from 0.5.1 to 0.5.3

- The files `LATEX2HTMLDIR/styles/german.perl` and `LATEX2HTMLDIR/styles/makeidx.perl` have been fixed so that they are consistent with changes in the main script.

- A problem with disappearing spaces after some equations and other environments has been fixed.

[24] http://cbl.leeds.ac.uk/nikos/tex2html/previous-versions
[25] http://cbl.leeds.ac.uk/nikos/tex2html/doc/Changes.txt

7.2 Changes from v0.4 to v0.5.1

Navigation Panel The navigation panel (see section 3.6) is now fully configurable and much better looking.

Automatic Signatures The signatures at the bottom of each page are now constructed using `nslookup` (Thanks to Alberto Accomazzi <alberto@cfa.harvard.edu>)

Numerous fixes including...

- "dead" `next_page` buttons when the `-split` option was used, and
- internal **LaTeX**2HTML markers appearing in section titles.

7.3 Changes from v0.3.1 to v0.4

Accents and Special Characters LaTeX accent commands, special characters and accent commands defined in `german.sty` are translated to equivalent ISO-LATIN-1 characters when that is possible (partly thanks to code supplied by Franz Vojik <vojik@de.tu-muenchen.informatik>).

Auto-loading of Style-Specific Code The translator now supports a mechanism for including Perl code extensions (see Section 2.2.1) which are specific to particular style files.

This mechanism will help to keep the core script smaller as well as make it easier for others to contribute and share solutions on how to translate specific style files. The current distribution includes the files `german.perl`, `french.perl`, `html.perl` and `makeidx.perl`.

Installation and Testing To make it easier to install the translator for more than one user there is now a configuration file (`LATEX2HTMLDIR/latex2html.config`) which contains installation specific information and default values for the various options. This makes it unnecessary to have a `HOME/.latex2html-init` for each user. If a user does have a `HOME/.latex2html-init` then this will be loaded after `LATEX2HTMLDIR/latex2html.config`.

A new Perl script (`install-test`) is now available with the distribution which will install the translator, and then perform some tests on the availability of some external programs giving, appropriate warnings.

Navigation Panel Extensions The following new links have been added to the navigation panel which appears in each page (where this is appropriate):

- a link to the next "logical" page (this allows reading each page in the same order as with a paper-based version - as opposed to structure navigation)

- a link to the previous "logical" page
- a link to the table of contents
- a link to the index

HTML Style File Extensions The HTML style file **`html.sty`** now includes definitions of new environments for:

Inclusion of Raw HTML A new LaTeX environment **`rawhtml`** allows arbitrary HTML tags to be included in a LaTeX document. This is useful for taking advantage of HTML+ features as they become available (e.g. interactive forms). The HTML commands are ignored when producing the DVI version of the document.

Conditional Text The new environments **`latexonly`** and **`htmlonly`** allow their contents to appear only in final DVI or in the HTML version of a document respectively.

A new command **`hyperref`** can be used to specify the way in which cross-references should be shown in the DVI version and the HTML version of the document.

Right Justification of Equations The translator adds enough whitespace at the beginning of each equation bitmap to push it to the right-hand margin. The width of each line can be set in the configuration file using the variable `LINE_WIDTH`.

Inlined Images All inlined images are now generated by calling **`latex`** and **`dvips`** only once at the end of the conversion process (but each generated image still has to be filtered individually). Also, a warning is given if an image cannot be converted.

Numeric Labels The original (LaTeX generated) numeric labels are used instead of the arrow navigation icon for cross-references where possible. To do this the translator uses the information in the **`aux`** file which is generated by LaTeX. If the **`aux`** file is out of date a warning is given.

New Options

`-auto_navigation` This puts a navigation panel at the top of each page as usual. But if the page exceeds a user-settable number of words (the default is 450 words) then a navigation panel is also placed at the end of the page. This option is active by default.

`-index_in_navigation` Adds a link to the index in the navigation panel.

`-contents_in_navigation` Adds a link to the table of contents in the navigation panel.

`-next_page_in_navigation` Adds a link to the next "logical" page in the navigation panel.

-previous_page_in_navigation Adds a link to the previous "logical" page in the navigation panel.

-bottom_navigation This puts a navigation panel at the bottom of each page.

-top_navigation This puts a navigation panel at the top of each page (the default).

-show_section_numbers This allows each section (page) title to be numbered in the same way that it would be numbered by LaTeX. This requires an up to date `aux` file (generated by running LaTeX) and that each title is unique. If the `aux` file is not up to date then a warning is given. Unfortunately if the title contains inlined images the numbering for that title will be lost.

-reuse This allows images generated during previous invocations of the translator to be "reused" without going through the initial interactive session. The same behavior is obtained by setting the variable `REUSE` to 1 (the default) in the configuration file. Note that images which may depend on contextual information (e.g. numerical labels) cannot be reused and are always re-generated.

Bug Fixes and Minor Changes

- The links from the table of contents in single page documents (created using the option `-split 0`) is now working as expected.
- Newlines are translated to the
 tag.
- Fixed some problems in dealing with new command macros.
- `texexpand` now respects commented `input` and `include` commands. Also (thanks to Franz Vojik <vojik@de.tu-muenchen.informatik>), it now looks in subdirectories when expanding files.
- The ALIGN attribute of inlined images is now BOTTOM instead of TOP.
- Fixed problem with the environment variable `TEXINPUTS`.
- Added some support for optional user-defined labels (bullets) in list environments.

More details on all the changes are available in the file `LATEX2HTMLDIR/Changes`.

7.4 Changes from v0.3 to v0.3.1

These changes are mostly due to patches contributed by Robert S. Thau <rst@ai.mit.edu>:

- Nested environments with the same name are now dealt with properly.
- Commands that are passed to LaTeX for processing which have environments in their arguments (e.g. a `parbox` command which an `itemize` environment as an argument) are now processed correctly. A general mechanism for users to specify the syntax of commands that should be passed to LaTeX is described in Page 8.
- Fixed a problem with recognizing the `special` command.
- Fixed bug in the generation of the index.

7.5 Changes from v0.2 to v0.3

Image Recycling Images for equations, tables, figures, special characters etc. generated by the translator are recognised during subsequent runs. The user is then asked whether old images should be reused (or whether the old images should be deleted and regenerated). This offers tremendous improvements in speed after an initial successful translation.

Cross-References Between (Local or Remote) Documents
Cross-references between two or more documents (possibly on remote locations) can be established via symbolic labels which are independent of the physical realisation of these documents.

Such cross-references will be maintained with a simple re-translation[26] even after one or more of the documents have been broken into different physical parts or moved.

The mechanism is based on the new commands (see Section 3.2) `externallabels` and `externalref` which are an extension of the simple `label-ref` pairs.

New Options

`external_images` This provides hypertext links to where generated images (for equations, tables, figures etc) are stored externally. (The default is to "inline" generated images in the main body of the text.)

`ascii_mode` This switches all the navigation icons to their ascii equivalents. Also generated images are stored externally as with the `external_images` option above. The `ascii_mode` option makes documents more portable as it allows them to be viewed on browsers that do not support inlined images.

[26] This is true for documents under the same server but for remote documents a little more is required (see the example in Section3.2.1)

Special Command Style File A style file `html.sty`[27] is now included in the distribution. This contains the definitions (syntax) of some special LaTeX commands mainly for providing external hypertext links. This should be included in LaTeX files that use the any hypermedia extensions (see Section3).

Minor Changes and Bug Fixes

- Equations, equation arrays and theorems are now numbered correctly even when they are individually passed to LaTeX for processing.
- Fixed problem with `label` commands appearing in section headings.
- Fixed problems with verbatim environments, bibliography items, generated file names, the options `nolatex` and `no_navigation`, the `thanks command`, the name of an HTML tag (<HEAD>),
- The appearance of footnotes has been improved.
- Some inconsistencies in the `pstoxbm` and `pstogif` scripts have been fixed.
- Parts of the documentation have been rewritten, restructured and some new sections have been added.

The following were contributed by Robert S. Thau <rst@ai.mit.edu>:

New Texexpand This fixes problems with the standard version and handles the inclusion of style files which need to processed by the translator. Appropriate modifications to the main script were also made to work with new version of `texexpand`.

Handling of Raw TeX Added support for simple raw TeX commands such as `special` and simple instances of `def`[28]. For the messier cases, the definition is scooped up and moved to the preamble. This allows the translator to handle the simple, but nonstandard, postscript figure inclusion macros as well as an awful lot of other stuff done with gratuitous `defs`.

More Options

`dont_include` This can be used to specify style files that should not be included in the translation.

`allbitmaps` The default format for generated images of equations and special commands is still XBM but the format of figures and tables has been changed to GIF. This reduces the space required to store the images (XBM's are much larger GIFs) but GIFs do not look as

[27] http://cbl.leeds.ac.uk/nikos/tex2html/doc/html.sty.txt

[28] Where "simple" is defined roughly as "they could have used `newcommand`, but didn't".

good on color monitors. This option allows all images to be generated as XBMs.

Other Changes

- Added support for nested math mode expressions and general list environments in order to handle *the document from Hell!!!*.
- Fixed problems in the translation of bibitems, the substitution of macro definitions, and the processing of unrecognized commands in the preamble.

7.6 Changes from v0.1.1 to v0.2

- Added a command line option to switch off the navigation links at the top of each page.
- The navigation icons are now part of the distribution.
- Added a customizable separator between the main body of the text in a page and the child links from that page.
- The order of the navigation keywords at the top of each page is the same as that of the navigation icons.
- The arguments of `verbatim` environments are now translated into fixed width fonts.
- A warning is given if a `bbl` (bibliography) file is needed but not found.
- The installation is now (mostly) done by setting variables in just one file.
- The `pstoxbm` script now uses environment variables set in the initialization file.
- Fixed bug in translating sequences of special HTML characters (e.g. &, <, etc.)
- Fixed bug in the handling of the `$$`-form of the dislay math environment.
- Fixed bug in the handling of the *-forms of environments.
- Added sections on how to embed hyperlinks in a LaTeX document (see Page 9) and on how to extend the translator (see Page 7) in the documentation.

7.7 Changes from v0.1 to v0.1.1

- Fixed bug about empty lines being inserted in environments that cannot tolerate them (e.g. `math`).
- Changed the format of inlined images coming back from LaTeX from GIF to XBM. This looks better on grayscale and color monitors.
- Fixed problem with commands being passed on to LaTeX after their arguments had been translated (this affects the commands `psfig`, `fbox`, `framebox`, and `parbox`).

8 Known Problems

Here are some of the problems of the current version:

Correctness and Efficiency The translator cannot be guaranteed to perform as expected. Several aspects of the implementation need optimization and improvement. Apart from possible bugs the translator may place heavy demands on your resources.

Unrecognized Commands and Environments Unrecognized commands are ignored and any arguments are left in the text. Unrecognized environments are passed to LaTeX and the result is included in the document as one or more inlined images.

Cross-references References in environments that are passed to LaTeX for processing (e.g. a `cite`, or a `ref` command), are not processed correctly. `label` commands are handled correctly.

Order Sensitive Commands Commands which affect global parameters during the translation, and are sensitive to the order in which they are processed may not be handled correctly. In particular, counter manipulation (e.g. `newcounter, setcounter, stepcounter`, etc) commands may cause problems.

Index The translator generates its own index by saving the arguments of the `index` command. The contents of the `theindex` environment are ignored.

New Definitions
New definitions (`newcommand`, `newenvironment`, `newtheorem` and `def`), will not work as expected if they are defined more than once. Only the last definition will be used throughout the document.

Scope of declarations and environments If the scope of a declaration or environment crosses section boundaries, then the output will not be as expected, because each section is processed independently.

9 Troubleshooting

Here are some curable symptoms:

It just stops Check the style files that you are using. It is likely that you are using a style file which contains raw TEX commands. In such a case start **LaTeX**2HTML with the option `-dont_include` <`style file name`>. Alternatively, add the name of the style to the variable `DONT_INCLUDE` in your `HOME/.latex2html-init` file. If you don't have such a file then create one and add the lines:

```
$DONT_INCLUDE = "$DONT_INCLUDE" . ":<style file name>";
1;  # This must be the last line
```

Another reason why **LaTeX**2HTML might stop is that the LATEX source file itself contains raw TEX commands. In this case you may put such comands inside a `latexonly (see Section` 3.4) environment.

Perl cannot parse the `latex2html` script Update your Perl to patch level 36. You can check which version of Perl you are using by invoking Perl with the `-v` option. Earlier versions of Perl than that shown above have caused problems due to tighter control over syntax.

It crashes (dumps core) as soon as it starts Update your Perl to patch level 36. You can check which version of Perl you are using by invoking Perl with the `-v` option.

While you wait for your technical support people to upgrade Perl you could try invoking Perl from within **LaTeX**2HTML with the `-d` (debug) option. Then, when **LaTeX**2HTML starts, it will immediately fall into the Perl debugger. To continue just press `c <CR>`.

`dvips` complains about incorrect arguments Please upgrade at least to version 5.516. **LaTeX**2HTML now invokes `dvips` with different arguments so that it generates a different postscript file for each page of a LATEX document (containing all the equations, tables, etc). The reason for using a single LATEX file is so that figures, tables and equations are numbered correctly.

Cannot convert postscript images which are included in the LATEX file It is likely that the macros you are using for including postscript files (e.g. `epsffile`) are not understood by **LaTeX**2HTML . To avoid this problem enclose them in an environment which will be passed to LATEX anyway e.g.

```
\begin{figure}
\epsffile{<postscript file name>}
\end{figure}
```

Another reason why this might happen is that your shell environment variable `TEXINPUTS` is undefined. In this case you have to use full pathnames for included postscript files (even when the postscript files are in the same directory as the LaTeX source file).

Some of the inlined images are in the wrong places This happens when any one of the inlined images is more than a page (paper page) long. It will be fixed.

Unacceptable quality of converted postscript images There are some inelegant ways to avoid this if the postscript image is a bitmap. Later versions of **LaTeX**2`HTML` will try and make use of the original postscript files instead of converting them to `GIF` format...

The bibliographic references are missing Run `latex` and then `bibtex` on the original source file in order to generate a `bbl` file. **LaTeX**2`HTML` requires a `bbl` in order to generate the references.

The labels of figures, tables or equations appearing in cross references are wrong This can happen if you have used any figures, tables, equations or any counters in conditional text i.e. in a `latexonly` or a `htmlonly` environment. Try to avoid using commands which generate automatic labels for cross-referencing in such environments (e.g. the `caption` command).

Problems when producing the DVI version If you are using any of the new LaTeX commands which are defined in the `html.sty` file make sure that `html.sty` file is included e.g. as one of the optional arguments to the `documentstyle` command.

Of course you also have to make sure that LaTeX knows where the html.sty file is, either by putting it in the same place as the other style files on your system, or by changing your TEXINPUTS shell environment variable[29].

Some of the fonts are translated incorrectly There is a fault in way the LaTeX scoping rules have been interpreted in **LaTeX**2`HTML` . Consider this:

```
\tt fixed-width font.
\begin{something}
nothing here
\end{something}
default font.
```

[29] If don't know how to do either of these things, copy (or link) html.sty to the directory of your LaTeX document...

When processed by LaTeX, the effect of the `tt` command is delimited by the beginning of the environment "something" so that "default font" will appear in the default font. But **LaTeX**2HTML will not recognize "something" as a delimiter and "default font" will appear in the wrong font.

To avoid this problem until it is fixed you may delimit the scope of some commands explicitly using `{}`'s i.e.

```
{\tt fixed-width font}.
\begin{something}
nothing here
\end{something}
default font.
```

Cannot get it to generate inlined images for equations etc. Try a small test file e.g.

```
% image-test.tex
\documentstyle{article}
\begin{document}
Some text followed by \fbox{some more text in a box}.
\end{document}
```

You should see something like:

```
This is LaTeX2HTML version 0.4a (Wed Dec 1 1993) by Nikos Drakos,
Computer Based Learning Unit, University of Leeds.

OPENING /usr/cblelca/nikos/scripts/tex2html/tests/image-test.tex

Reading ....
Translating ...0/1.....1/1......
Generating images using latex ...
This is TeX, C Version 3.14t3
(12222_images.tex
LaTeX Version 2.09 <7 Dec 1989>


Generating postscript images using dvips ...
This is dvips 5.521 Copyright 1986, 1993 Radical Eye Software
' TeX output 1993.12.03:1050' -> 12222_image
(-> 12222_image001) <tex.pro>[1]
Initializing... done.
```

```
Ghostscript 2.6.1 (5/28/93)
Copyright (C) 1990-1993 Aladdin Enterprises, Menlo Park, CA.
  All rights reserved.
Ghostscript comes with NO WARRANTY: see the file COPYING for details.
GS>GS>Writing 12222_image001.ppm
GS>pnmcrop: cropping 119 rows off the top
pnmcrop: cropping 961 rows off the bottom
pnmcrop: cropping 208 cols off the left
pnmcrop: cropping 484 cols off the right

Doing section links .....
Done.
```

If there is a problem somewhere during the conversion from postscript to GIF you can try to do it manually so that you can find out where the problem is. Here is one way to do it (the `pstoppm.ps` file comes with the **LaTeX**2HTML distribution ...):

```
cblelca% latex image-test.tex
This is TeX, C Version 3.14t3
(image-test.tex
LaTeX Version 2.09 <7 Dec 1989>
(/usr/TeX/tex.lib/inputs//paper.sty
Document Style `paper' <28 Nov 89>.
(/usr/TeX/tex.lib/inputs//pap11.sty) (/usr/TeX/tex.lib/inputs//doublespace.sty)
(/usr/TeX/tex.lib/inputs//smaller.sty)) (/usr/TeX/tex.lib/inputs//psfig.sty
psfig/tex 1.9
)
No file image-test.aux.
[1] (image-test.aux) )
Output written on image-test.dvi (1 page, 652 bytes).
Transcript written on image-test.log.
cblelca% dvips -o image-test.ps image-test.dvi
This is dvips 5.519 Copyright 1986, 1993 Radical Eye Software
' TeX output 1993.11.12:1412' -> image-test.ps
<tex.pro>. [1]
cblelca% gs -dNODISPLAY pstoppm.ps
Initializing... done.
Ghostscript 2.6.1 (5/28/93)
Copyright (C) 1990-1993 Aladdin Enterprises, Menlo Park, CA.
  All rights reserved.
Ghostscript comes with NO WARRANTY: see the file COPYING for details.
GS>(image-test) ppm1run
Writing image-test.ppm
```

```
GS>quit
cblelca% pnmcrop image-test.ppm >image-test.crop.ppm
pnmcrop: cropping 61 rows off the top
pnmcrop: cropping 110 rows off the bottom
pnmcrop: cropping 72 cols off the left
pnmcrop: cropping 72 cols off the right
cblelca% pbmtoxbm image-test.crop.ppm >image-test.xbm
```

It cannot do slides, memos, etc, ... Be patient!

10 Sample Converted Documents

A comparison between the paper-based version (or postscript[30]) and the hypertext equivalent[31] of the documentation on the LaTeX to HTML translator shows most of its features. Both of the above documents were generated from the same LaTeX source file[32] The following is a small selection of other documents that have been processed locally through the translator and have some interesting features.

- A description of the **IRC** (Internet Relay Chat).[33] This contains plenty of tables, preformatted data, lots of cross-references, footnotes and an index.

- A scientific paper[34] containing a large number of numbered equations. The cross-references to the equations from the main text were generated using the usual LaTeX commands.
- Interactive graphical programming environments and software construction[35]. A collection of quotes and figures with a large bibliography, many cross-references and an index.
- A classic paper on Lisp.[36]

These are a couple of interesting examples of the uses of **LaTeX**2HTML :

- The Electronic Book[37] produced by the Computational Science Education Project.

[30] http://cbl.leeds.ac.uk/nikos/tex2html/doc/latex2html.ps
[31] http://cbl.leeds.ac.uk/nikos/tex2html/doc/latex2html/latex2html.html
[32] http://cbl.leeds.ac.uk/nikos/tex2html/doc/latex2html.tex
[33] http://cbl.leeds.ac.uk/nikos/tex2html/examples/IRCprimer1.1/IRCprimer1.1.html
[34] http://cbl.leeds.ac.uk/nikos/tex2html/examples/physics/physics.html
[35] http://cbl.leeds.ac.uk/nikos/tex2html/examples/concepts/concepts.html
[36] http://cbl.leeds.ac.uk/nikos/tex2html/examples/good-bad-win/good-bad-win.html
[37] http://compsci.cas.vanderbilt.edu/csep.html

- The MIT Transit Project Documentation[38]. Some of the reports here use hypertext links in their LaTeX bibliographies.

- A document on electronic submission of manuscripts to[39] an IEEE journal, recommending the use of **LaTeX**2HTML .

[38]http://www.ai.mit.edu/projects/transit/tn-cat.html
[39]http://www.research.att.com/esubmit/esubmit.html

11 Figures, Tables and Arbitrary Environments

Here are some examples of how the translator handles figures, tables and other environments. This is taken from [1] (page 64):

gnats	gram	\$13.65
	each	.01
gnu	stuffed	92.50
emur		33.33
armadillo	frozen	8.99

Table 1: A sample table

Here are some some automatically numbered right-justified equations

$$\Phi_{l+1,m,n} = (\Phi + h\frac{\partial\Phi}{\partial x} + \frac{1}{2}h^2\frac{\partial^2\Phi}{\partial x^2} + \frac{1}{6}h^3\frac{\partial^3\Phi}{\partial x^3} + \ldots)_{l,m,n} \tag{1}$$

with some gratuitously áccënted text in between them.

$$\frac{\Phi_{l+1,m,n} - 2\Phi_{l,m,n} + \Phi_{l-1,m,n}}{h^2} + \frac{\Phi_{l,m+1,n} - 2\Phi_{l,m,n} + \Phi_{l,m-1,n}}{h^2} + \frac{\Phi_{l,m,n+1} - 2\Phi_{l,m,n} + \Phi_{l,m,n-1}}{h^2} = -I_{l,m,n}(v) \tag{2}$$

12 General License Agreement and Lack of Warranty

This software is distributed in the hope that it will be useful but **without any warranty**. The author(s) do not accept responsibility to anyone for the consequences of using it or for whether it serves any particular purpose or works at all. No warranty is made about the software or its performance.

Use and copying of this software and the preparation of derivative works based on this software are permitted, so long as the following conditions are met:

- The copyright notice and this entire notice are included intact and prominently carried on all copies and supporting documentation.
- No fees or compensation are charged for use, copies, or access to this software. You may charge a nominal distribution fee for the physical act of transferring a copy, but you may not charge for the program itself.

- If you modify this software, you must cause the modified file(s) to carry prominent notices (a Change Log) describing the changes, who made the changes, and the date of those changes.
- Any work distributed or published that in whole or in part contains or is a derivative of this software or any part thereof is subject to the terms of this agreement. The aggregation of another unrelated program with this software or its derivative on a volume of storage or distribution medium does not bring the other program under the scope of these terms.

This software is made available **as is**, and is distributed without warranty of any kind, either expressed or implied. In no event will the author(s) or their institutions be liable to you for damages, including lost profits, lost monies, or other special, incidental or consequential damages arising out of or in connection with the use or inability to use (including but not limited to loss of data or data being rendered inaccurate or losses sustained by third parties or a failure of the program to operate as documented) the program, even if you have been advised of the possibility of such damages, or for any claim by any other party, whether in an action of contract, negligence, or other tortious action.

The LaTeX to HTML translator is written by Nikos Drakos, Computer Based Learning Unit, University of Leeds, Leeds, LS2 9JT.

13 Credits

Several people have contributed suggestions, ideas, solutions, support and encouragement. Some of these are Roderick Williams, Ana Maria Paiva, Jamil Sawar and Andrew Cole here at the Computer Based Learning Unit.

The idea of splitting LaTeX files into more than one components linked with hyperlinks was first implemented in Perl by Toni Lantunen at CERN. Thanks to Robert Cailliau <cailliau@cernnext.cern.ch> of the WorldWide Web Project also at CERN for giving me access to the source code and documentation (although no part of the original design or the actual code has been used).

Robert S. Thau <rst@edu.mit.ai> has contributed the new version of `texexpand`. Also, in order to translate the *document from hell* (!!!) he has extended the translator to handle `def` commands, nested math-mode commands, and has fixed several bugs.

The `pstoxbm` and `pstogif` scripts are modified versions of the `pstoepsi` script written by Doug Crabill <dgc@cs.purdue.edu>. They use the `pstoppm.ps` postscript program originally written by Phillip Conrad (Perfect Byte, Inc.) and modified by L. Peter Deutsch (Aladdin Enterprises).

The idea of using existing symbolic labels to provide cross-references between documents was first conceived during discussions with Roderick Williams <rodw@cbl.leeds.ac.uk>.

Eric Carroll <eric@ca.utoronto.utcc.enfm> suggested providing a command like `hyperref`. Franz Vojik <vojik@de.tu-muenchen.informatik> provided the basic mechanism for handling foreign accents. The `-auto_navigation` option was based on an idea by Todd <little@com.dec.enet.nuts2u>. Axel Belinfante <Axel.Belinfante@cs.utwente.nl> provided the code in the `makeidx.perl` file as well as numerous suggestions and bug reports.

Verena Umar <verena@edu.vanderbilt.cas.compsci> (Computer Science Education Project[40]) has been a very patient tester of some early versions of **LaTeX2HTML** and many of the current features are a result of her suggestions.

Many others, too many to mention, have contributed bug reports, fixes, and other suggestions. Keep them coming!

References

[1] Leslie Lamport. *LATEX User's Guide & Reference Manual.* Addison-Wesley Publishing Company, Inc., 1986.

[40] http://compsci.cas.vanderbilt.edu/csep.html

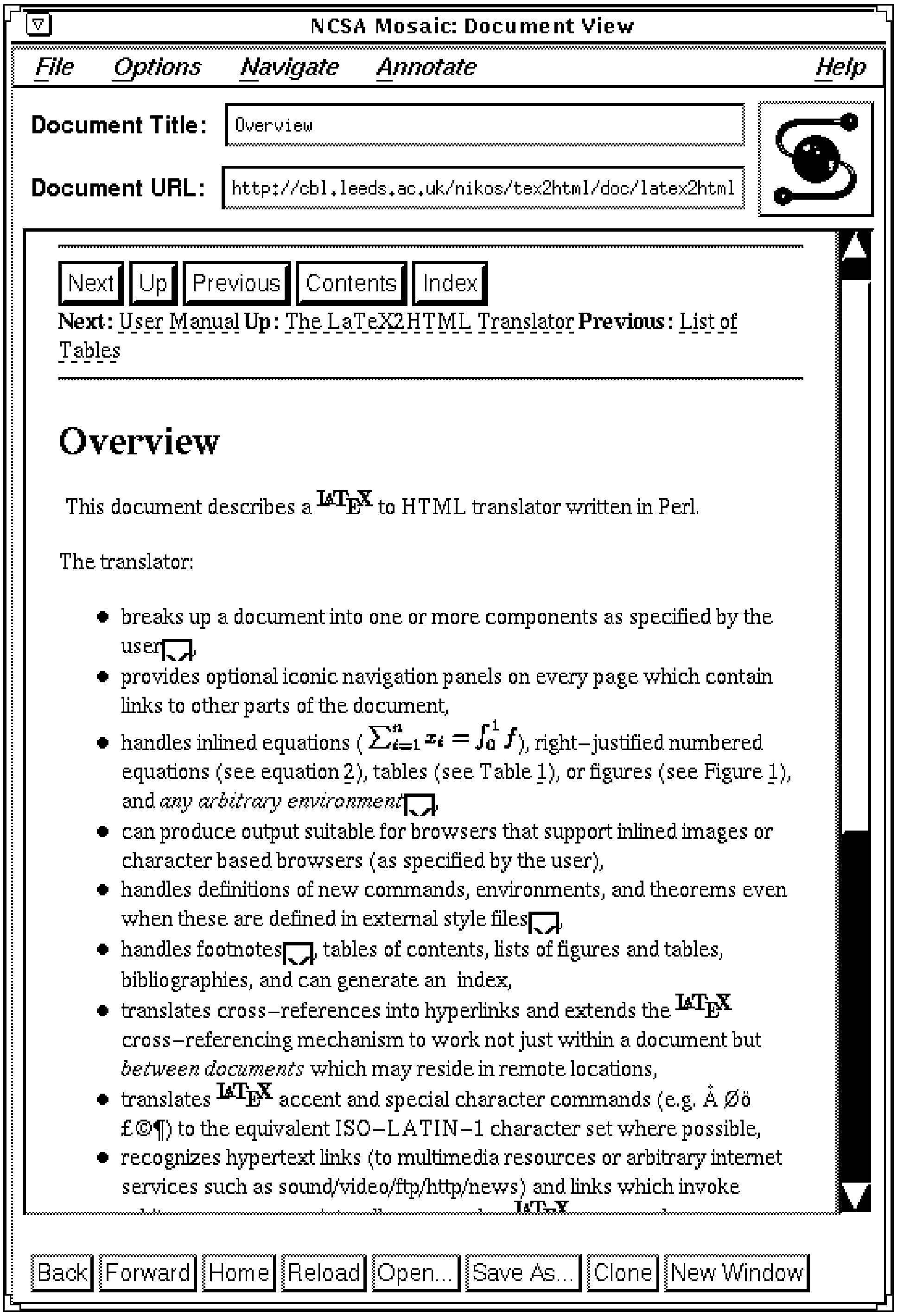

Figure 1: A sample figure

Index

bugs 25
conditional text 13
copyright 33
cross-rederences 13
cross-references 25
debugging 26
efficiency 25
electronic book 30
electronic forms 11
examples 30
externallabels 10
externalref 10
figures 32
fixes 26
HTML+ 11
html.sty 9
htmladdimg 9
htmladdnormallink 9
htmlonly 12
hyperlinks 9
hyperref 13
hypertext extensions 9
index 25
index 4
inlined equations 4
ISO-LATIN-1 4
labels.pl 10
latexonly 12
man page 5
navigation panel 14
new definitions 25
numbered equations 32
options 5
overview 4
problems 25
problems 26
rawhtml 11
requirements 16
scientific paper conversion 30
source code 15
tables 32
texexpand 16
unrecognized commands 25