The LaTeX-access manual

Alastair Irving ¡alastair.irving@sjc.ox.ac.uk¿
Robin Williams ¡rmw205@exeter.ac.uk¿
Daniel Dalton ¡daniel.dalton10@gmail.com¿
Nathaniel Schmidt ¡nathanieljsch@westnet.com.au¿
Stefan Moisei ¡vortex37@gmail.com¿

May 17, 2011

Contents

Latex-access – provides a blind person with a more efficient means of interacting with LaTeX for mathematical and scientific documents.

Chapter 1

Introduction

The latex-access project is designed to provide a realtime translation of a line of LaTeX in to braille, concentrating on the Nemeth code, whichh can be read on a refreshable braille display. This will greatly improve the ease of use of LaTeX to blind mathematicians and scientists. The project also translates the current line into english speech which is easier to listen to than LaTeX source.

Note that this project is largely aimed at people wishing to read LaTeX using a refreshable braille display and/or speech synthesisor[B, and people who will probably want to edit LaTeX documents. For example, as a university student, I receive my worksheets in LaTeX format, and produce my work using LaTeX. Using the latex_access package, I am able to get a fairly good translation of the question and then an on-the-fly translation of my work as I produce it. If you are not concerned with editing LaTeX documents and simply want a braille translation of an entire laTeX document, then this project is not for you.

There is also a very low traffic mailing list, which is worth subscribing to if you have any queries, problems, suggestions or ideas. All current developers are subscribed to this list and are very willing to assist. To subscribe send an email with the word “subscribe” in the subject to: latex-access-devel-request@lists.sourceforge.net. To post to the list send emails to latex-access-devel@lists.sourceforge.net

1.1 Purpose

It is widely thought that LaTeX is a good system for a blind mathematician or scientist to use to create and read scientific documents, as it is a linear code and so the user does not have to perceive two-dimensional concepts, such as fractions and column vectors. By reading this linear code, a blind person can take in and understand scientific documents in the same way that a sighted person would do by studying a printed document. It should be noted that normally, laTeX is just a source from which documents are converted in to an attractive-looking, typeset document that can be printed or viewed on screen, often in a .pdf, .dvi or .ps format. For various technical reasons, documents in such formats are currently inaccessible with current screen-reading technology. The best current solution therefore is not to concern ourselves with documents in these formats, but rather to read and interpret the LaTeX source code itself.

1.2 Reading a LaTeX document

It is entirely possible to read a LaTeX document simply by reading the LaTeX source itself. This however, is often a time-consuming and pain-staking process, and it is often not particularly nice to read. For example, the LaTeX source for the quadratic formula is
$$x=\frac{-b\pm\sqrt{b2̂-4ac}}{2a}$$
It is therefore the aim of the project to translate a line of LaTeX in to a line of Niemeth braille code, which can be read using a refreshable braille display. The project also aims to provide an audible translation of the LaTeX source which will be output through current screen-reading technology.

1.3 Current features

latex-access currently contains the following features.

- Translation of several mathematical expressions from LaTeX to Niemeth braille. These include, but are not confined to:
  - Translation of fractions, both numerical and algebraic.
  - Translation of trigonometric functions and hyperbolic functions.
  - Translation of powers, including square roots.
  - Translation of expressions used in calculus, including partial derivatives.
  - Translation of two component and three component column vectors, not in to Niemeth braille format but in to a row vector so that they can be read on a single line braille display.

  - Translation of several mathematical symbols, such as the Greek letters.
  - Many commands used to create a visually attractive document are either translated or ignored, often it is not necessary to see some formatting commands.

- Translation of several of the above to audible speech.

- A matrix browser feature to enable easier reading of larger matrices in LaTeX, see the description below.

- Support for custom defined LaTeX commands.

Chapter 2

Installation

2.1 Obtaining the source

The package is hosted by svn at, `https://latex-access.svn.sourceforge.net/svnroot/latex-access`.

This link will take you to a web interface of the svn tree, but you'll probably want to checkout the code so you can install it. If you run windows see section 2.1.2, and if you run linux see section 2.1.1.

2.1.1 Linux

Under Linux, the standard subversion command line client works well. This can usually be installed on debian based distros by running
apt-get install subversion
# (as root)

Then type
svn co https://latex-access.svn.sourceforge.net/svnroot/latex-access latex-access

This will check the package out into the directory latex-access.

In future feel free to run "svn up", to pull the latest updates from the server. (from within the latex-access directory).

2.1.2 Windows

On windows the "Tortoise SVN" client works well.
`http://tortoisesvn.tigris.org/`

Once you have checked out the sourcecode from
`https://latex-access.svn.sourceforge.net/svnroot/latex-access`
continue with the installation process.

Note, you should periodically pull the latest updates from the server to get the latest and greatest features of latex-access.

2.2 Installing the package

Currently latex-access interacts with three different programs. Therefore, the installation process varies slightly.

- Jaws for windows – most developed front-end to the project, and allows Jaws to interact with latex-access within most editors. See section 2.2.1.
- Emacs/emacspeak – Provides both Braille and speech access when working with latex under the emacs editor. This means anyone using a Linux system can use the project. Usage under Linux is very stable, but there are still a few little things to be done, see the README in the emacs branch for details. Currently has not been tested with the windows emacs version. See section 2.2.2 for installation instructions.
- NVDA – provides latex-access for usage with most editors. Under heavy development still, but it provides many features of latex-access. See section 2.2.3 for installation instructions.

For specific notes on each individual front-end it is worth looking at the READMES in their respective branches. You can also find details about contacting the developer of the specific front-end from the readme.

2.2.1 Jaws

To install the scripts on a machine running windows and Jaws version 5 or higher, do the following.

1. Download the latest stable release of version 2 of the Python software from `http://www.python.org` and install it.

   Note: LaTeX-access has not been tested with python 3 and probably won't work with it.

2. Download the pywin32 package from `https://sourceforge.net/projects/pywin32` ensuring you have the correct file to match your version of python. Install this.

3. Create a directory on the c: drive named latex_access. (The name of the directory or it's level of depth in the filesystem does not matter, however choosing a directory with a name easy to remember and not too deep in the filesystem makes the next step easier.

4. Obtain the latest version of the scripts, (although you've presumably already done this). For details of how to obtain the latest version please see section 2.1.

5. Extract the files and folder in the zip file to the latex_access folder you created earlier. To do this, either simply run the zip file with an application such as winrar and copy everything to the folder, or extract the files and folder to the relevant location by right clicking (or equivalent) on the file within windows explorer.

6. Copy the files from within the jaws folder to the folder where your jaws scripts are located. These are usually in a path such as
   c:\documents and settings\%username%\application data\freedom scientific\jaws\%jaws
   or
   c:\Users\% username% \AppData\freedom scientific\jaws\%jaws version%\settings\enu
   (This can be reached using the explore jaws submenu of the jaws menu in your start menu).

7. Now open a command prompt by going to run, and typing "cmd". Switch to the latex_access directory by typing 'cd %directory%' for example, type "cd c:\latex_access"

8. Type
   "latex_access_com.py"
   you should hear "latex_access registered". Now register the matrix processor object by typing
   "matrix_processor.py" You should hear a similar message. Exit the command prompt by typing exit.

   Note: the object here is to run the specified python files with python, so the above will only work if python is the default program associated with .py files. If it is not then try python latex_access_com.py and if this fails then use the full path to your python installation, for example
   c:\python26\python.exe latex_access_com.py
   then repeat with matrix_processor.py

9. Finally, open the confignames.ini file in your jaws scripts folder using a text editor such as notepad. After the line which reads [ConfigNames] enter a new line as described below.

As many different applications can be used to read or write LaTeX, the scripts have a generic name latex.j**. Therefore, we use this file to add an alias, so that many different applications can be used without needing to change the name of each Jaws script to match the executable name of the application used to write LaTeX. So for example, as a user of winedt, I have "winedt=latex" If you use notepad to write and read LaTeX, you should add "notepad=latex" etc. If you are one of multiple users on the computer, you may have to search to find the confignames.ini file. The file will be found within the same file structure, within the all users (or something similar) directory, within the documents and settings directory.

Note: with older versions of jaws you could just enter the lines at the bottom of the file, but in more recent versions the .ini file contains more sections and the lines must be inserted in the [ConfigNames] section.

10. Load the relevant program for writing/reading LaTeX, and press ctrl+m to initialise the scripts. Repeat this keystroke to turn them off. You should be able to navigate the document and listen to audible speech output, as well as reading the mathematical translation on the braille display.

2.2.2 Emacs/emacspeak

I assume you have downloaded or checked out the package from subversion. If not, please see section 2.1

Installing emacs/emacspeak support for latex-access.

1. - A. Automated installation script. I have built an installation script titled "setup.py" found in the emacs branch of this svn checkout.

     Invoking with no arguments will install to $\tilde{/}$.emacs, otherwise if you know what your doing and use an init file somewhere else, feel free to invoke the script as follows:
python setup.py ¡path-to-init-file¿

     Most users who are happy with the default ($\tilde{/}$.emacs), may just invoke the script as follows:
python setup.py

     You may as well say yes to the prompt (y), to byte-compile emacs-latex-access... This should improve the speed slightly. If it fails or you do not want to byte compile the script can handle this fine. (It modifies your init file according to whether it could byte-compile or not.)

Continue following the prompts until the script has finished.

- B. Manual installation. If you used the script, then skip to step 2. Add the following to ~/.emacs. Change /path/to/svnroot to the actual path to your svn local checkout. Eg. the directory containing the bulk of the *.py files and the general readme.txt file. Below lines should be appended to your .emacs.

  ```
  ; Emacs latex-access:
  (setq latex-access-path ”/path/to/svnroot”)
  (load (concat latex-access-path ”/emacs/emacs-latex-access.el”))
  (add-hook ’LaTeX-mode-hook ’latex-access-speech-on) ; comment this if you don’t have emacspeak
  (add-hook ’LaTeX-mode-hook ’latex-access-braille-on) ; comment if you don’t have Braille
  ; End emacs Latex-access.
  ```

  Note: Keep this structure if you wish to have the uninstall script work with this installation.

  See the comments if you don’t have Braille or emacspeak present.

  If you wish to use a byte-compiled file, for improved speed, replace .el with .elc in the line:
  (load (concat latex-access-path ”/emacs/emacs-latex-access.el”))

  To byte-compile the emacs-latex-access.el file, do:
  emacs -batch -f batch-byte-compile emacs-latex-access.el
  (From the emacs branch of the svn checkout)

2. Append to the PYTHONPATH environment variable the directory /path/to/svnroot, replacing /path/to/svnroot with the actual path to your svn checkout of latex-access if you used the manual installation. Otherwise copy and paste the export path output by the installation script. This is usually done by an export line in .bash_profile. Ensure the variable is set before starting emacs.

3. Please install pymacs. On debian/ubuntu you may do:
   sudo apt-get install pymacs
   Otherwise follow the instructions provided at: `http://pymacs.progiciels-bpi.ca/`

4. Restart emacs! Now emacs should communicate correctly with latex-access.

5. If you wish to place any settings in your .emacs for latex-access, place them under the line
   ; end latex-access

   - A. To set how many linesabove the currently selected line should also be translated add this line anywhere below
     ; end latex-access
     (setq latex-access-linesabove 1)
     This would Braille the current line and the one above, change 1 to whatever setting you desire. 0 is just the current line.

2.2.3 NVDA

Requirements

To use this plugin, you will need to install the following software/libraries:

- NVDA, version 2011.1 or later. The NVDA website can be found at `http://www.nvda-project.org`, and the download page can be found here.
- The latest build of version 2 of the python programming language. Official website: `http://www.python.org/`.
- The python for windows extentions, which can be downloaded at `http://sourceforge.net/projects/pywin32/`. This package is required to registor the latex-access COM object and the matrix.
- The latex-access scripts. You probably already have these, but if not see section 2.1. (and be sure to remember which directory you checked out to)

Registering the COM object and installing the globalPlugin.

Assuming you have followed the steps above and installed everything you need, you can install the plugin and register the COM object by doing the following:

For registering the COM object:

- Type windows+r to go into the run dialog, then type cmd and press enter.
- type “cd”, then type in the location where you installed the latex-access scripts. E.G.
  cd C:\latex-access\
- to register the COM object for latex-access translation, type:
  latex_access_COM.py
- To register the matrix, type:
  matrix_processor.py
  In both cases, you should hear iether the message “latex_access registered,” or “latex_access_matrix” registered” which indicates that what you just did was successful. If not, then something is obviously wrong. Remember that if you change the location of the latex-access scripts, you have to register the COM objects again.
- Note: the object here is to run the specified python files with python, so the above will only work if python is the default program associated with .py files. If it is not then try :
  python latex_access_COM.py
  however this only works if you have python in your path variable. if this fails then use the full path to your python installation, for example :
  c:\python26\python.exe latex_access_COM.py
  then repeat with matrix_processor.py.
- type “exit” (without the quotes) in your currently open cmd shell.

To install the plugin

- Go to the nvda subdirectory in the latex-access folder, select latex_access.py and copy it, either by pressing ctrl+C or right clicking on it and choosing “copy.”
- Now you need to paste this file in the appropriate folder so that the plugin will be recognised by NVDA. Depending on your distribution/installation, the globalPlugins directory will be in one of the following places by default:
  - Installed version of NVDA:

    - For Windows XP:
      C:\Documents and Settings\username\Application Data\NVDA\globalPlugins\
    - for Windows Vista/7:
      C:\Users\username\AppData\Roaming\NVDA\globalPlugins\
  - Portable archive or running from source:
    DirectoryOfArchive/distribution\userConfig\globalPlugins\

- Once latex_access.py is pasted, launch NVDA by what ever means possible, and if the error sound doesn't play, everything should go smoothly.

A note about edit fields:

If you use a keystroke such as ctrl+m, it will only work if you are in an edit field, and one that has accessible text enabled at that. You will either get errors, or a different response from an application if you try to use the latex_access global plugin in edit fields that are not native controls. I'm afraid that's just the way windows works. I know for a fact that notepad works with the plugin quite well, but in an editor such as EdSharp it does not.

Chapter 3

General usage

Chapter 4

More advance usage

Chapter 5

Extra components

Chapter 6

For developers

Appendix A

Jaws – default key bindings

Appendix B

NVDA – default bindings

Appendix C

Emacs – functions and bindings

Appendix D

Future features