# Plover: Open Source Stenography Software

Joshua Harlan Lifton

November 12, 2010

## 1 Quick Start

### 1.1 Installation

The latest release of *Plover* can be downloaded from `https://launchpad.net/plover` as a compressed `.tar.gz` archive that must be extracted. Among the extracted files will be a README.txt file. Follow the directions in this file for installing and running *Plover*.

### 1.2 Starting the Program

Once installed, *Plover* can be run from either the menu (`Applications->Universal Access->Plover`) or from a terminal window (`plover` and then `ENTER`). If a stenotype machine other than the default is used, set the machine type in the configuration dialog.

### 1.3 Configuration

*Plover* comes with a default configuration that should work for most setups that use the Microsoft Sidewinder X4 keyboard as the stenotype machine. *Plover* can be configured by right-clicking on the *Plover* task bar icon and selecting the `Configure...` option. Within the configuration dialog, the user can select which stenotype machine to use, which dictionary file to use, and whether or not strokes and translations should be logged to a file. *Plover* will need to be restarted before any configuration changes take effect.

## 2 Stenography Dictionary

At the core of stenography is a dictionary that translates stroke combinations into English words and phrases.

## 2.1 Dictionary Format

*Plover* expects the stenography dictionary file to be in JSON format and each dictionary entry within the file to be in a specific variant of RTF/CRE format. A simple dictionary entry looks like:
`"stroke": "Some English text",`
A dictionary entry for more than one stroke looks like:
`"stroke1/stroke2/stroke3": "Some more English text",`
Meta commands are surrounded by { and }, like:
`"stroke1/stroke2": "{^opolis}",`
which means append "opolis" to the previous word without inserting a space.
Meta commands and regular text can be mixed, like:
`"stroke1/stroke2": "{.}Thus, we can see that",`
which means append a period and the number of spaces after a period and the text "Thus, we can see that".
Spaces between meta commands and regular text don't count, so the previous example is the same as:
`"stroke1/stroke2": "{.} Thus, we can see that",`
Furthermore, since the {.} meta command will also capitalize the next word, the previous example is also the same as:
`"stroke1/stroke2": "{.} thus, we can see that",`
There can be more than one meta command in a translation, like:
`"stroke1/stroke2": "{,}like{,}"`
Since ", {, and } are all special dictionary format characters, the character sequences \", \{, and \}, should be used within a translation, like:
`"stroke1/stroke2": "{^.\" }{-|}",`
which will append ." and a space to the previous word and then capitalize the next word.

## 2.2 Available Meta Commands

- Sentence stops: {.}, {!}, {?} all end a sentence with the corresponding punctuation, insert a space and capitalize the next letter.
- Sentence breaks: {,}, {:}, {;} all break a sentence with the corresponding punctuation.
- Simple suffixes: {^ed}, {^ing}, {^er}, {^s} applies basic orthographic rules to append the corresponding prefix to the most recent word to form the past tense, present progressive, noun, or plural, respectively.
- Capitalize: {-|} capitalizes the next letter.

- Glue flag: {&X} will connect the string X without a space to any adjacent (previous or following) strings that also have glue flags.
- Attach flag: {^X}, {X^}, {^X^} will connect the string X without a space to the previous, next, or both previous and next, respectively, words, unless X is one of the simple suffixes (ed, ing, er, s), in which case the simple suffix rules take precedence.
- Key combinations: {#X} will execute the key combination described by X. See below for details.

### 2.3 Arbitrary Key Combinations

Meta commands of the form {#X} are interpreted as a sequence of keyboard keys pressed and released in sequence and/or simultaneously. In sequence key presses are separated by a single space. Simultaneous key presses are denoted by parentheses surrounding keys pressed while another key is held down. For example:
`Alt_L(Tab)`
will emulate the action of holding down the left Alt key while pressing and releasing the Tab key and then releasing the left Alt key. Parentheses can be nested, as in:
`Control_L(Shift_L(minus minus minus)`
which would emulate holding down the left Control key, then holding down the left Shift key, and then pressing the minus (-) key three times before releasing the left Shift and then the left Control keys.
Below is the list of all legal keys that can be used to form key combination commands.

- 0 1 2 3 4 5 6 7 8 9
- a b c d e f g h i j k l m n o p q r s t u v w x y z
- A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
- Alt_L Alt_R Control_L Control_R Hyper_L Hyper_R Meta_L Meta_R Shift_L Shift_R Super_L Super_R
- Caps_Lock Num_Lock Scroll_Lock Shift_Lock
- Return Tab BackSpace Delete Escape Break Insert Pause Print Sys_Req
- Up Down Left Right Page_Up Page_Down Home End

- F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F16 F17 F18 F19 F20 F21 F22 F23 F24 F25 F26 F27 F28 F29 F30 F31 F32 F33 F34 F35
- L1 L2 L3 L4 L5 L6 L7 L8 L9 L10
- R1 R2 R3 R4 R5 R6 R7 R8 R9 R10 R11 R12 R13 R14 R15
- KP_0 KP_1 KP_2 KP_3 KP_4 KP_5 KP_6 KP_7 KP_8 KP_9 KP_Add KP_Begin KP_Decimal KP_Delete KP_Divide KP_Down KP_End KP_Enter KP_Equal KP_F1 KP_F2 KP_F3 KP_F4 KP_Home KP_Insert KP_Left KP_Multiply KP_Next KP_Page_Down KP_Page_Up KP_Prior KP_Right KP_Separator KP_Space KP_Subtract KP_Tab KP_Up
- ampersand apostrophe asciitilde asterisk at backslash braceleft braceright bracketleft bracketright colon comma division dollar equal exclam greater hyphen less minus multiply numbersign parenleft parenright percent period plus question quotedbl quoteleft quoteright semicolon slash space underscore
- AE Aacute Acircumflex Adiaeresis Agrave Aring Atilde Ccedilla Eacute Ecircumflex Ediaeresis Egrave Eth ETH Iacute Icircumflex Idiaeresis Igrave Ntilde Oacute Ocircumflex Odiaeresis Ograve Ooblique Otilde THORN Thorn Uacute Ucircumflex Udiaeresis Ugrave Yacute
- ae aacute acircumflex acute adiaeresis agrave aring atilde ccedilla eacute ecircumflex ediaeresis egrave eth iacute icircumflex idiaeresis igrave ntilde oacute ocircumflex odiaeresis ograve oslash otilde thorn uacute ucircumflex udiaeresis ugrave yacute ydiaeresis
- cedilla diaeresis grave asciicircum bar brokenbar cent copyright currency degree exclamdown guillemotleft guillemotright macron masculine mu nobreakspace notsign onehalf onequarter onesuperior ordfeminine paragraph periodcentered plusminus questiondown registered script_switch section ssharp sterling threequarters threesuperior twosuperior yen
- Begin Cancel Clear Execute Find Help Linefeed Menu Mode_switch Multi_key MultipleCandidate Next PreviousCandidate Prior Redo Select SingleCandidate Undo
- Eisu_Shift Eisu_toggle Hankaku Henkan Henkan_Mode Hiragana Hiragana_Katakana Kana_Lock Kana_Shift Kanji Katakana Mae_Koho Massyo Muhenkan Romaji Touroku Zen_Koho Zenkaku Zenkaku_Hankaku

### 2.4 Editing the Dictionary

To edit the dictionary file, issue the following command in a terminal window:

```
gedit $HOME/.config/dict.json
```

or a comparable command with the location of whatever dictionary file *Plover* is configured to use. In case of a character encoding error in `gedit`, do the following:

1. Click the character encoding drop-down list.
2. Select Add or Remove...
3. Select Western ISO-8859-1, also known as latin-1.
4. Click the Add button.
5. Click the OK button.
6. Select ISO-8859-1 from the character encoding drop-down list.
7. Click the Retry button.

## 3 Further Resources

Other than this document, there are several resources for getting help with *Plover*:

- mailing list: `http://groups.google.com/group/ploversteno`
- blog: `http://plover.stenoknight.com`
- IRC channel: `#plover` on `irc.freenode.net` or `http://webchat.freenode.net/?channels=#plover`
- download and development: `http://launchpad.net/plover`
- general information: `http://stenoknight.com/plover`