Fenrir User Manual for Slint users

Fenrir is a modern command line screen reader written in Python3.
It has a modular structure, a flexible based driver model, is highly configurable and easy to customize and extend.

Credit and intended audience

This document is just a customization for Slint of the genuine Fenrir User Manual written by Christian Hempfling aka Chrys, main developer of Fenrir.

It has been adapted to its intended audience: end users of Fenrir on Slint where it is already installed, thus concentrates on its setting and usage. You will find more information about its features, installation and how customize and troubleshoot it and contribute to its development on the Fenrir Git repository.

Getting started with Fenrir

Fenrir is a screen reader, that reads aloud the text displayed on the screen and allows to review it, like espeakup and speechd-up also shipped in Slint.
In Slint it is mostly used in console mode, and uses the synthesizers and voices provided by Speech Dispatcher.
Keyboard shortcuts are bound to commands to read the screen, review it and perform other actions like modify the settings on the fly, switch languages, copy and paste text, provide miscellaneous information.

Configuration file

The permanent settings are recorded in the configuration file /etc/fenrirscreenreader/settings.conf, that is heavily commented.
These settings are described in this user guide.

To start Fenrir once type as root:

```
service start fenrir
```

To enable auto start on system boot type as root:

```
speak-with fenrir
```

and confirm that you want to have it started at boot time.

If you are using Fenrir for the first time you may want to take a look at these resources:

- Keybindings

- Tutorial Mode

Keybindings

Two series of key bindings are provided:

1. the Desktop layout uses a numeric keypad (here abbreviated as keypad).
2. the Laptop layout for keyboards that do not have a numeric keypad.

The layout is set in the configuration file in the section Keyboard.

Fenrir Key

The Fenrir Key is for invoking screen reader commands. Fenrir can utilize more than one FenrirKey at the same time. By default the following keys are used:

1. Insert
2. KeyPad Insert
3. Meta (Super, Windows)

Script Key

To invoke “Scripts” the Script Key is mandatory. The shortcut is encoded in the filename of the script. See Scripting

By default the Script Key is Compose (the key between AltGr and Ctrl at the right of the space bar).

For instance, pressing Script Key + L switches between the languages possibly listed in the configuration file. Feel free to write you own scripts to expand Fenrir features and share them!

Desktop Layout

Shortcut	Command
General	
CTRL	shut up (interrupts speech)
FenrirKey + Q	quits fenrir
Review modes	
FenrirKey + KeyPad dot	exit review
FenrirKey + KeyPad 9	reviews bottom
FenrirKey + KeyPad 7	reviews top
KeyPad 8	reviews current line
KeyPad 7	reviews previous line
KeyPad 9	reviews next line
FenrirKey + KeyPad 4	reviews line beginning
FenrirKey + KeyPad 6	reviews line ending

Shortcut	**Command**
General	
FenrirKey + KeyPad 1	reviews line first character
FenrirKey + KeyPad 3	reviews line last character
FenrirKey + Alt + 1	presents first line
FenrirKey + Alt + 2	presents last line
KeyPad 5	reviews current word
KeyPad 4	reviews previous word
KeyPad 6	reviews next word
FenrirKey + Shift + KeyPad 5	reviews current word phonetic
FenrirKey + Shift + KeyPad 4	reviews previous word phonetic
FenrirKey + Shift + KeyPad 6	reviews next word phonetic
KeyPad 2	reviews current char
KeyPad 1	reviews previous char
KeyPad 3	reviews next char
FenrirKey + Shift + KeyPad 2	reviews current character phonetic
FenrirKey + Shift + KeyPad 1	reviews previous character phonetic
FenrirKey + Shift + KeyPad 3	reviews next character phonetic
FenrirKey + CTRL + KeyPad 8	reviews up
FenrirKey + CTRL + KeyPad 2	reviews down
FenrirKey + I	indent curr line
FenrirKey + KeyPad 5	current screen
FenrirKey + KeyPad 8	current screen before cursor
FenrirKey + KeyPad 2	current screen after cursor
<Unbound>	cursor read to end of line
Tutorial mode	
FenrirKey + H	toggle tutorial mode
Information	
FenrirKey + T	Announce time
FenrirKey + T +T	Announce date
FenrirKey + CTRL + <n>	clear bookmark number <n>
FenrirKey + Shift + <n>	set bookmark number <n>
FenrirKey + <n>	read bookmark number <n>
KeyPad dot	cursor position
<Unbound>	cursor column
<Unbound>	cursor line number
Screen Interaction	
FenrirKey + CTRL + Shift + C	clears clipboard
FenrirKey + Home	first clipboard
FenrirKey + End	last clipboard
FenrirKey + PageUp	previous clipboard
FenrirKey + PageDown	next clipboard
FenrirKey + Shift + C	current clipboard
FenrirKey + C	copy marked text to clipboard
FenrirKey + V	paste clipboard contents
FenrirKey + P	import clipboard from file

Shortcut	Command
General	
FenrirKey + Alt + Shift +C	export clipboard to file
FenrirKey + CTRL + Shift + X	remove marks
FenrirKey + X	set mark
FenrirKey + Shift + X	announce marked text
<Unbound>	export clipboard to X
Window mode	
FenrirKey + KeyPad Slash	set window application
2 * FenrirKey + KeyPad Slash	clear window application
KeyPad Plus	read last incoming
Quick settings: toggle or switch	
Compose + L	switch language
FenrirKey + F3	toggles sound
FenrirKey + F4	toggles speech
KeyPad Enter	temporarily disables speech
FenrirKey + CTRL + P	toggles punctuation level
FenrirKey + RightBrace	toggle auto spell check
FenrirKey + Backslash	toggles output
FenrirKey + CTRL + E	toggles emoticons
FenrirKey + KeyPad Enter	toggles auto read
FenrirKey + CTRL + T	toggles auto time
FenrirKey + KeyPad ASTERISK	toggles highlight tracking
Quick Settings: speech and sound	
FenrirKey + Up	increase speech volume
FenrirKey + Down	decrease speech volume
FenrirKey + Right	increase speech rate
FenrirKey + Left	decrease speech rate
FenrirKey + Alt + Right	increase speech pitch
FenrirKey + Alt + Left	decrease speech pitch
FenrirKey + Alt + Up	increase sound volume
FenrirKey + Alt + Down	decrease sound volume
FenrirKey + CTRL + Up	increase Alsa volume
FenrirKey + CTRL + Down	decrease Alsa volume
Spell checker	
FenrirKey + S	spell check
2 * FenrirKey + S	add word to spell check
FenrirKey + Shift + S	removes word from spell check
FenrirKey + Backspace	forward keypress

Laptop Layout

Shortcut	Command
General	
FenrirKey + H	toggle tutorial mode
CTRL	shut up (interrupts speech)

Shortcut	Command
General	
Review modes	
FenrirKey + Shift + O	reviews bottom
FenrirKey + Shift + U	reviews top
FenrirKey + I	reviews current line
FenrirKey + U	reviews previous line
FenrirKey + O	reviews next line
FenrirKey + Shift + J	reviews line beginning
FenrirKey + Shift + L	reviews line ending
FenrirKey + CTRL + J	reviews line first character
FenrirKey + CTRL + L	reviews line last character
FenrirKey + Alt + 1	presents first line
FenrirKey + Alt + 2	presents last line
FenrirKey + K	reviews current word
FenrirKey + J	reviews previous word
FenrirKey + L	reviews next word
FenrirKey + CTRL + ALT + K	reviews current word phonetic
FenrirKey + CTRL + ALT + J	reviews previous word phonetic
FenrirKey + CTRL + ALT + L	reviews next word phonetic
FenrirKey + comma	reviews current character
FenrirKey + M	reviews previous character
FenrirKey + dot	reviews next character
FenrirKey + CTRL + ALT + comma	reviews current character phonetic
FenrirKey + CTRL + ALT + M	reviews previous character phonetic
FenrirKey + CTRL + ALT + dot	reviews next character phonetic
FenrirKey + CTRL + I	reviews up
FenrirKey + CTRL + comma	reviews down
FenrirKey + Slash	exit review
FenrirKey + Shift + dot	cursor position
FenrirKey + I +I	indent curr line
FenrirKey + Shift + K	current screen
FenrirKey + Shift + I	current screen before cursor
FenrirKey + Shift + comma	current screen after cursor
<Unbound>	cursor read to end of line
Tutorial mode	
FenrirKey + H	toggle tutorial mode
Information	
FenrirKey + T	Announce time
FenrirKey + T +T	Announce date
FenrirKey + CTRL + <n>	clear bookmark number <n>
FenrirKey + Shift + 1	set bookmark bumber <n>
FenrirKey + 1	bookmark number <n>
<Unbound>	cursor column
<Unbound>	cursor line number
Screen Interaction	

Shortcut	Command
General	
FenrirKey + CTRL + Shift + C	clears clipboard
FenrirKey + Home	first clipboard
FenrirKey + End	last clipboard
FenrirKey + PageUp	previous clipboard
FenrirKey + PageDown	next clipboard
FenrirKey + Shift + C	current clipboard
FenrirKey + C	copy marked text to clipboard
FenrirKey + V	paste clipboard contents
FenrirKey + F5	import clipboard from file
FenrirKey + Alt + Shift +C	export clipboard to file
FenrirKey + CTRL + Shift + X	remove marks
FenrirKey + X	set mark
FenrirKey + Shift + X	announce marked text
<Unbound>	export clipboard to X
Window mode	
FenrirKey + CTRL + 8	set window application
2 * FenrirKey + CTRL + 8	clear window application
FenrirKey + Semicolon	read last incoming
Quick settings: toggle or switch	
Compose + L	switch language
FenrirKey + F3	toggles sound
FenrirKey + F4	toggles speech
FenrirKey + Enter	temporarily disables speech
FenrirKey + Shift + CTRL + P	toggles punctuation level
FenrirKey + RightBrace	toggle auto spell check
FenrirKey + Shift + Enter	toggles output
FenrirKey + Shift + E	toggles emoticons
FenrirKey + Enter	toggles auto read
FenrirKey + CTRL + T	toggles auto time
FenrirKey + Y	toggles highlight tracking
FenrirKey + Q	quits fenrir
FenrirKey + T	Announce time
FenrirKey + T +T	Announce date
Quick Settings: speech and sound	
FenrirKey + Up	increase speech volume
FenrirKey + Down	decrease speech volume
FenrirKey + Right	increase speech rate
FenrirKey + Left	decrease speech rate
FenrirKey + Alt + Right	increase speech pitch
FenrirKey + Alt + Left	decrease speech pitch
FenrirKey + Alt + Up	increase sound volume
FenrirKey + Alt + Down	decrease sound volume
FenrirKey + CTRL + Up	increases Alsa volume
FenrirKey + CTRL + Down	decreases Alsa volume

Shortcut	Command
General	
Spell checker	
FenrirKey + S	spell check
FenrirKey + S +S	add word to spell check
FenrirKey + Shift + S	removes word from spell check
FenrirKey + Backspace	forward keypress

General

quit Fenrir

Just stops fenrir.

shut up

Interrupt the current spoken.

Review Modes

Fenrir provides a virtual cursor, with it you can navigate all over the screen without changing the text cursor.
Using the review feature will open the review mode automatically.
The review cursor always starts from the text cursor. Attention: after using the review mode, the review cursor will stay open until you use the `exit review` shortcut.
Think when using clipboard operations and similar. The review cursor is always preferred over the text cursor.
Fenrir sounds a bell sound if the used review command jumps to another line or end of screen.

exit review

You can leave the review mode by pressing the `exit review` shortcut.

review bottom

Set the review cursor to first column in the last line.

review top

Set the review cursor to the first column in the first line

review current line

Set the review cursor to the beginn of the the current line and review it.

review previous line

Set the review cursor to the previous line and review it.

review next line

Set the review cursor to the next line and review it.

review line beginning

Set the review cursor to the begin of the current line

review line ending

Set the review cursor to the end of the current line

review line first character

Set the review cursor the first char (that is not space) in the current line and review it.

review line last character

Set the review cursor the last char (that is not space) in the current line and review it.

review current word

Sets the review cursor to the beginning of the current word and review it.

review previous word

Sets the review cursor to the beginning of the previous word and review it.

review next word

Sets the review cursor to the beginning of the next word and review it.

review current word phonetic

Sets the review cursor to the beginning of the current word and spell it phonetic.

review previous word phonetic

Sets the review cursor to the beginning of the previous word and spell it phonetic.

review next word phonetic

Sets the review cursor to the beginning of the next word and spell it phonetic.

review current character

Does not change the review cursor. Just announce the current char.

review previous character

Sets review cursor to the previous column and review it

review next character

Sets review cursor to the next column and review it

review current character phonetic

Does not change the review cursor. Just announce the current char phonetic.

review previous character phonetic

Sets review cursor to the previous column and announce the char phonetic.

review next character phonetic

Sets review cursor to the next column and announce the char phonetic.

review up

Set the review cursor in the same column one line above the current one and review it.

review down

Set the review cursor in the same column one line below the current one and review it. A mark defines a point of origin or end to prepare to copy or paste a block of text.
Examples where you need marks are:

- copy to clipboard
- set window application
- set bookmark 1 - X

Set mark

How to set a mark:

1. navigate with review or textcursor to the position you want to set the mark. Attention: if a review cursor is set, that is the prefered. If you want to use text cursor, be sure that you are not in review mode.
2. press shortcut for `set mark`

you can set two marks (begin and end). Some commands allow some simpler usecases just using the whole line if only one mark is set. you may want to try this out.

Get text between marks

To get the text that is currently between your marks press shortcut for `marked text`.

Remove Marks

You can remove all current marks by pressing the shortcut for `remove marks`. Changing the screen also removes the marks.

Screen Interaction

Fenrir provides several methods to interact with the current screen.

forward keypress

This just forwards the next shortcut to the screen Fenrir shortcut or not. This is useful if the currently pressed shortcut is also in use by Fenrir.

Clipboard

Fenrir provides a clipboard with multible items represented by a list. You navigate throught the list and paste the selected clipboard.

copy marked to clipboard

To copy something to the clipboard you need to set one or two marks. if you set one mark, the text between the mark and your current cursor is copied to clipboard. Setting two marks just copies the text between the marks into the clipboard. If you copy something it is always placed as the first item on your clipboard.

clear clipboard

You can remove all items from the current clipboard by `clear clipboard` functionality.

first clipboard

This moves quick to the first item of the clipboard.

last clipboard

This moves quick to the last item of the clipboard.

previous clipboard

Go to previous item in the clipboard.

next clipboard

Go to next item on the clipboard.

read current clipboard

Read the content of the current item of the clipboard.

paste clipboard

Pass whatever item is currently selected by first, last, prev or next clipboard commands. if no special clipboard is selected the (last copied) is used.

export clipboard to file

This allows you to export the current clipboard to a configurable filepath. This is useful to share the clipboard with a graphical desktop.

import clipboard from file

Import a clipboard from a configurable file. This is useful to share the clipboard with a graphical desktop.

Quick Settings: toggle or switch

Fenrir provides shortcuts to change settings temporarily and on the fly without the need to permanently change the `settings.conf` file.

switch language

switch between the languages listed in the line beginning with langages= in `setting.conf` during run time. The next language after the current one will be used.

toggle braille

Enables and disables Braille. This is not persistent stored in your `settings.conf` but during run time.

toggle sound

Enables and disables sound. This is not persistent stored in your `settings.conf` but during run time.

toggle speech

Enables and disables speech. This is not persistent stored in your `settings.conf` but during run time.

disable speech temporarily

Disables the speech until next key press. it might be useful if you want to listen to music or similar. As soon as a key is pressed it is going to be enabled again.

toggle punctuation level

Cycle between all available punctuation levels. This is not persistent stored in your `settings.conf` but during run time.

toggle auto spell check

Enables and disables automatic spellchecker (when typing). This is not persistent stored in your `settings.conf` but during run time.

toggle emoticons

Enables and disables emoticons. This is not persistent stored in your `settings.conf` but during runtime.

toggle output

Enables and disables all output at once (sound, Braille, speech). This is not persistent stored in your `settings.conf` but during run time.

toggle auto read

Enables and disables what is automatically spoken. This is not persistent stored in your `settings.conf` but during run time.

toggle auto time

Enables and disables auto time functionality. This is not persistent stored in your `settings.conf` but during run time.

toggle highlight tracking

Enables and disables highlight tracking. This is not persistent stored in your `settings.conf` but during run time.

Quick Settings: speech and sound

increase speech volume

Increase the volume of the speech. This is not persistent stored in your `settings.conf` but during runtime.

decrease speech volume

Decrease the volume of the speech. This is not persistent stored in your `settings.conf` but during runtime.

increase speech rate

Increase the rate of the speech. This is not persistent stored in your `settings.conf` but during runtime.

decrease speech rate

Decrease the rate of the speech. This is not persistent stored in your `settings.conf` but during runtime.

increase speech pitch

Increase the pitch of the speech. This is not persistent stored in your `settings.conf` but during runtime.

decrease speech pitch

Decrease the pitch of the speech. This is not persistent stored in your `settings.conf` but during runtime.

increase sound volume

Increase the volume of the sound. This is not persistent stored in your `settings.conf` but during runtime.

decrease sound volume

Decrease the volume of the sound. This is not persistent stored in your `settings.conf` but during runtime.

Window Mode

Fenrir supports window mode, a window is a partial area of the screen.

Create Window

To create a window you need to do the following:

1. set a beginning mark (as the start of the window)
2. set an end mark (where the window should end)
3. press `set window application` shortcut.

Now Fenrir ignores anything outside of the window.

Remove Window

You can remove the window by pressing `the clear window application` shortcut. Now Fenrir will read everything on the screen again.

Tracking Modes

Different types of tracking are currently supported See section Focus in `settings.conf` for more information.

Cursor Tracking

This follows the text cursor. This is the typical way an application works. This is used by:

- almost any shell such as (Bash, Zsh, sh)
- vim
- nano
- emacs
- mutt
- tintin++

Highlight Tracking

In some applications there are no text cursors. In those applications cursor changes are represented by different colors or attributes (underlined or bold). This mode tracks and announces these changes for you. This is used by:

- wifi-menu
- dialog
- alpine

Tutorial Mode

Fenrir provides a Tutorial mode. When you enter tutorial mode, screen reader commands are intercepted and explained instead of executing them. `Arrow up` and `Arrow Down` let you navigate through a list of all available commands with shortcuts and description. Pressing escape leaves the tutorial mode.

Information

Time

Announces the current Time.

Date

Announces the current Date.

Bookmarks

Bookmarks provide quick access to part of the screen without the need to navigate to the area. By default Fenrir provides 10 bookmarks. Those can be set and accessed via shortcut. This is useful for status lines or other information where the position does not change.

set Bookmark <n>

You need to set the bookmark first. For that you have to set one or two lines for use.

1. Set marks (one or two)
2. press shortcut for `set bookmark` <n>. <n> represents the number 1 - 10 (set bookmark 0 for the tenth bookmark).

read Bookmark <n>

If a bookmark is set you can access the area just by pressing the `bookmark` <n> shortcut. <n> represents the number 1 - 10. Bookmarks are dynamic. That means the content changes with the screen.

clear Bookmark <n>

to remove a bookmark just press the `clear bookmark` <n> shortcut. <n> represents the number 1 - <n>. Afterward the bookmark is no longer available.

cursor position

You can get information about the current cursor and its position by using the “cursor position” functionality.

indent current line

Announce the current indent level of the current line. It represents the number of trailing spaces of the line.

current screen

Reads all the current screen from the beginning to the end.

current screen before cursor

Reads current screen from the beginning of the screen to the current cursor position.

current screen after cursor

Read anything after current cursor position to the end.

cursor read to end of line

Read from the current cursor position to the end of the current line.

cursor column

Read the current X position of a cursor (column of the current line).

cursor line number

Read the current Y position of a cursor (line number).

present first line

Reads just the first line. this is maybe useful for status information.

present last line

Presets the last line. This is maybe useful for status information.

last incoming

Repeat the last automatically incoming text.

Input

Echo

Fenrir provides different methods of echoing content:

- Word: Will speak each word after you push space
- Character: speak any letter you type on the screen
- Delete Character: speaks the character prior to the cursor when you push backspace

Silence on Key press

Spellchecker

Fenrir has a built-in spellchecker, it can invoke automatically while typing or be called by a shortcut. Commands to add or remove the current word to the dictionary are included. As using the spellchecker is enhanced usage. You will need dictionary aspell-<language>. See section General in `settings.conf` for more information.

spell check

Invokes the spellcheck on the word that contains the Review or text cursor.

add word to spell check

Adds the word under the Review or Text cursor to the dictionary.

removes word from spell check

Removes the word under the Review or Text cursor from the dictionary.

Announcements

Emoticons

If you want to replace ":)" emoticons with "smile" in speech you can use this feature. It can be toggled on or off. You can define emoticons in a dictionary, please see Emoticon Dictionary. See section General in `settings.conf` to see how to enable or disable this feature.

Time

Announce the time at periodical increments, To track the time easily. You can define 2 different ways of time announcements.

1. periodic
2. on fix minutes

The settings proposed below can be implemented editing in the configuration file

Example periodic, every 20 minutes "delaySec=20":

```
[time]
enabled=True
presentTime=True
presentDate=True
delaySec=20
onMinutes=
announce=True
interrupt=False
```

Example on fix minutes in an hour. example every quarter "delaySec=0" and "onMinutes=00,15,30,45":

```
[time]
enabled=True
presentTime=True
presentDate=True
#delaySec is respected before onMinutes so it need to be set to 0
delaySec=0
onMinutes=00,15,30,45
announce=True
interrupt=False
```

Promoted List

Promoted Lists are a nice feature if you are away from your computer or performing more longer tasks. you can define a list of words which you want to hear a sound icon for after a period of inactivity. Example if the word "Chrys" appears after 120 Seconds of inactivity:

```
[promote]
enabled=True
inactiveTimeoutSec=120
list=Chrys
```

See section Promote in `settings.conf` for more information.

Dictionary

You can make use of different kinds of built-in dictionary's. A dictionary has a name and list of keys and values separated by :===: Example to be set in the configuration file

```
[customDict]
Chrys:===:Chrys is cool
lollipop:===:lolli
```

that means that every instance “chrys” is displayed, speech will say Chrys is cool. “lollipop” is spoken as “lolli”. Before making changes to a dictionary we recommend making a backup of your current dictionary in case future updates overwrite your local changes.

Punctuation

The default punctuation settings are recorded in thd directory /etc/fenrirscreenreader/punctuation/default.conf

Level

The punctuation level dict contains lists with “what punctuation is spoken in what level”. the default one looks like this:

```
[levelDict]
none:===:
some:===:.-$~+*-/\@
most:===:.,:-$~+*-/\@!#%^&*()[]}{<>;
all:===:!"#$%& \'()*+,-./:;<=>?@[\\]^_`{|}~
```

the level “none” has no values. so it should not speak any punctuation (sadly this is not respected by every TTS system) if “some” is the current level the following are spoken: dot dash dollar tilde plus star slash backslash at. same for most and all, you can add new levels. if you cycle punctuation levels they are recognized. the default punctuation level is set in settings.conf. The default is “some”.

Punctuation

The punctuation dictionary “[punctDict]” contains how the punctuation is spoken. Example:

```
[punctDict]
_:===:line
```

speaks an _ as “line”.

```
[punctDict]
_:===:underscore
```

speaks an _ as underscore. for question mark an ? is appended to the word that the TTS system can announce the question correctly.

Custom

The dict “[customDict]” is just for your own use, it just replace the key with the value without any special functionality. This might be used to fix incorrectly spoken words, make words more common, shorter or just for fun. :)

Emoticons

The Emoticons dictionary “[emoticonDict]” by default contains some emoticons. it can replace “:)” with “smile” or “XD” with “loool” Making chat more colorful. A nice feature with this dictionary is that you can toggle the substitution on or off during run time or in settings.conf. This is useful because while doing programming or other serious work you want to hear colons and perryns not smiles.

Configuration

You can configure Fenrir in the following places (ordered by priority):

1. Commandline Parameters `-o` see Set settings options
2. /etc/fenrirscreenreader/settings/settings.conf see Settings
3. hard coded defaults

Commandline Arguments

Set settings options

You can specify options that overwrite the setting.conf. This is done with `-o <list of options>` parameter. The list of options have the following syntax

```
fenrir -o "section#setting=value;section#setting=value"
```

For example changing the sound driver to gstreamer and disabling Braille

```
fenrir -o "sound#driver=gstreamerDriver;braille#enabled=False=False"
```

or change the debug level to verbose

```
fenrir -o "general#debugLevel=3"
```

You can find the available sections and variables here Settings See Syntax settings.conf syntax

settings.conf syntax

the syntax of the settings.conf is quite simple and similar to a “*.ini” file, there are 4 different elements.

1. Sections
2. Settings
3. Values
4. Comments

A comment starts with a # and is ignored by Fenrir.

```
# this is a comment
```

To group settings we have sections. A section can look like this:

```
[Section]
```

A setting looks like this:

```
settingName=Value
```

Example:

```
[sound]
# Turn sound on or off:
enabled=True
# Select the driver used to play sounds, choices are genericDriver and
gstreamerDriver.
# Sox is default.
driver=genericDriver
```

Settings

Sound

The sound is configured in section `[sound]`.

Turn sound on or off:

```
enabled=True
```

Values: on=`True`, off=`False`

Select the driver used to play sounds. The genericDriver using Sox is the default.

```
driver=genericDriver
```

Available Drivers:

- `genericDriver` using the generic driver, for Fenrir <1.5 just use `generic`
- `gstreamerDriver` using the gstreamer, for Fenrir <1.5 just use `gstreamer`

These are the pack of sounds used for sound icons.

```
theme=default
```

By default we ship two sound packs.

1. `default` opus encoded, for newer Sox versions
2. `default-wav` wav encoded, just for compatibility

Sound packs are located at /usr/share/sounds/fenrirscreenreader/

Sound volume controls how loud the sounds for your selected sound pack are.

```
volume=1.0
```

Values: `0.0` is quietest, `1.0` is loudest.

Generic Driver

The generic sound driver uses shell commands for play sound and frequencies.

`genericPlayFileCommand` defines the command that is used to play a sound file.

```
genericPlayFileCommand=<your command for playing a file>
```

`genericFrequencyCommand` defines the command that is used playing frequencies.

```
genericFrequencyCommand=<your command for playing a frequence>
```

The following variables are substituted in `genericPlayFileCommand` and `genericFrequencyCommand`:

- `fenrirVolume` = the current volume setting
- `fenrirSoundFile` = the sound file for an sound icon
- `fenrirFrequence` = the frequency to play
- `fenrirDuration` = the duration of the frequency

Example genericPlayFileCommand (default)

```
genericPlayFileCommand=play -q -v fenrirVolume fenrirSoundFile
```

Example genericFrequencyCommand (default)

```
genericFrequencyCommand=play -q -v fenrirVolume -n -c1 synth fenrirDuration
sine fenrirFrequence
```

Speech

Speech is configured in section `[speech]`. Turn speech on or off:

```
enabled=True
```

Values: on=`True`, off=`False`

# Select speech driver, options are speechdDriver (default), genericDriver or espeakDriver: driver=speechdDriver #driver=espeakDriver #driver=genericDriver

This Selects the driver used to generate speech output.

We recommend top keep the default in Slint to benefit of all available synthesizers and associated voices

The rate selects how fast Fenrir will speak.

```
rate=0.5
```

Values: Range Minimum:`0.0` is slowest, Maximum:`1.0` is fastest.

Pitch controls the pitch of the voice.

```
 pitch=0.5
```

Values: Range Minimum:`0.0` is lowest, Maximum:`1.0` is highest.

A Pitch for capital letters can be set.

```
capitalPitch=0.9
```

Values: Range Minimum:`0.0` is lowest, Maximum:`1.0` is highest.

The Volume controls the loudness of the voice.

```
volume=1.0
```

Values: Range Minimum:`0.0` is quietest, Maximum:`1.0` is loudest.

The Module sets here selects the synthesizer to use by speech-dispatcher. If not set, the one set in

/etc/speech-dispatcher/speechd.conf will be used.
TO list the available modules, type:

```
spd-list -s
```

To be able to switch languages on the fly, set below a module that supports all languages between which you want to be able to switch.
To make this choice type:

```
spd-list
```

Example of module setting:

```
module=espeak-ng
```

Voice selects the voice you want to use. This setting won't hold if you switch languages on the fly (see below) or also set the language.
To find out which voices are available, for a given module, type:

```
spd-say -o <module> -L
```

For instance:

```
spd-say -o rhvoice -L
```

If you use the module espeak-ng or espak-ng-mbrola-generic , you can also run:

```
espeak-ng-list-voices
```

Example using the pico module:

```
voice=virginie
```

Languages lists the languages among which to switch pressing:

```
Compose+l
```

The Compose key is the one between AltrGr and Ctrl on the right of the space bar.
For the module in use, the languages available can be listed with typing:

```
spd-list -ls <module>
```

For instance:

```
spd-list -ls espeak-ng-mbrola-generic
```

Be careful: all languages listed below should be provided by the module in use, else switching languages on the fly won't be possible. Example using the pico module:

```
languages=en-GB fr-FR es-ES it-IT
```

You can also set the language to use by default, which doesn't need to be listed among the ones you might want to switch (but then you will need to restart fenrir to get it back).
It will supersede the voice that you might have also set. Example using the pico module:

```
language=de-DE
```

Read new text as it occurs autoReadIncoming=True Values: on=`True`, off=`False`

Screen

The settings for screens, (TTY, PTY) are configured in the `[screen]` section.

The driver to get the information from the screen:

```
driver=vcsaDriver
```

Available Drivers:

- `vcsaDriver` using the VCSA driver (for TTYs), for Fenrir <1.5 just use `vcsa`

The encoding of the screen

```
encoding=cp850
```

Values:`cp850` is used for Western languages like USA or Europe.

The driver updates Fenrir with changes on the screen.

```
screenUpdateDelay=0.05
```

Values: in Seconds

If you want Fenrir to not be active on any screen for various reasons. Maybe an X server or Wayland is running on that screen. You can make Fenrir ignore it or multiple screens seperated by `,` with:

```
suspendingScreen=
```

Values: Depends on driver:

- VCSA: the number of the TTY. TTY6 is 6.

Example ignore TTY1 and TTY2:

```
suspendingScreen=1,2
```

There is also the ability to let Fenrir auto detect screens that are running an X server. So Screens running an GUI can be ignored.

```
autodetectSuspendingScreen=True
```

Values: on=`True`, off=`False`

Keyboard

The settings for keyboard and input related configuration is located in the section `[keyboard]` of the `settings.conf` file.

Select the driver used for grabbing keybord input and for recieving shortcuts.

```
driver=evdevDriver
```

Values: Text, available Driver Available Drivers:

- `evdevDriver` uses the evdev input system of linux, for Fenrir <1.5 just use `evdev`

You can let Fenrir know about what input devices are to be used.

```
device=ALL
```

Values:

- `ALL` use all devices with key capabilities.
- `NOMICE` Exclude mices from handling.
- `<Device Name>` just use the device with the given name.

Gives Fenrir exclusive access to the keyboard and lets it control keystrokes. This is needed to intercept Fenrir related shortcuts.

```
grabDevices=True
```

Values: on=`True`, off=`False`

The following makes sense if you are using a second screenreader and want to have some hooked events. Fenrir ignores all shortcuts then.

```
ignoreShortcuts=False
```

Values: on=`True`, off=`False`

The current keyboard layout used for shortcuts.

```
keyboardLayout=desktop
```

Values: An absolute Path to a Keyboard definition file or a Filename without extension located in `/etc/fenrir/keyboard`

Announce characters while typing.

```
charEcho=False
```

Values: on=`True`, off=`False`

Announce deleted characters

```
charDeleteEcho=True
```

Values: on=`True`, off=`False`

Announce word after pressing space

```
wordEcho=False
```

Values: on=`True`, off=`False`

Interrupt speech on any keypress

```
interruptOnKeyPress=False
```

Values: on=`True`, off=`False`

You can filter the keys that speech should interrupt

```
interruptOnKeyPressFilter=
```

Values: (List) empty = all keys, otherwise interrupt with specified keys

The timeout that is used for double tap shortcuts

```
doubleTapTimeout=0.2
```

Values: Seconds

General

Overall settings can be configured from the section `[general]`.

Set the current debug level:

```
debugLevel=1
```

Values: off=0, error=1, warning=2, info=3

the current punctuation and dict file in use:

```
punctuationProfile=default
```

Values: Text, see available profiles in `/etc/fenrir/punctuation` or in `sourceTree/config/punctuation`

The current punctuation level in use:

```
punctuationLevel=some
```

Values: Text, See available levels in the used punctuation file.

Respect pause for punctuations:

```
respectPunctuationPause=True
```

Values: on=`True`, off=`False`

Add a pause on Line break:

```
newLinePause=True
```

Values: on=`True`, off=`False`

Specify the path where the clipboard should be exported to. See export clipboard to file. The variable `$user` is replaced by the current logged username.

```
clipboardExportPath=/tmp/fenrirClipboard
```

Values: Text, Systemfilepath

The number of available clipboards:

```
numberOfClipboards=10
```

Values: Integer, 1 - 999

Replace emoticons like :) or ;) with text insertions:

```
emoticons=True
```

Values: on=`True`, off=`False`

Define the current Fenrir keys:

```
fenrirKeys=KEY_KP0,KEY_META,KEY_INSERT
```

Values, Text list, separated by comma.

Define the current script keys:

```
scriptKey=KEY_COMPOSE
```

Values, Text list, separated by comma.

The time format to be used for (time command) output:

```
timeFormat=%H:%M:%P
```

Values: see python specification for datetime.strftime

The date format to be used for (date command) output:

```
dateFormat=%A, %B %d, %Y
```

Values: see python specification for datetime.strftime

Enable or Disable spellcheck whilst typing:

```
autoSpellCheck=True
```

Values: on=`True`, off=`False`

The use of the dictionary with spellcheck:

```
spellCheckLanguage=en_US
```

Values: Text, see aspell dictionary's.

Folder Path for your scripts “scriptKey” functionality:

```
scriptPath=/usr/share/fenrirscreenreader/scripts
```

Values: Text, Existing path on file system.

Override commands or create new ones without changing the Fenrir defaults:

```
commandPath=/usr/share/fenrirscreenreader/commands
```

Values: Text, Existing path on file system. Subfolders in commandPath are:

- `commands` = to create shortcut commands
- `onInput` = executed while typing
- `onScreenChange` = executed on change the screen (change from TTY4 to TTY6)
- `onScreenUpdate` = executed when the screen is captured

Focus

The configuration for basic focus is in the section `[focus]`. Follow the text cursor:

```
cursor=True
```

Values: on=`True`, off=`False`

Follow highlighted text changes (Highlight Tracking):

```
highlight=False
```

Values: on=`True`, off=`False`

Review

Configurations for the review mode are in the section `[review]`.

If “next word/ char” or “prev word/char” create a linebreak, announce it:

```
lineBreak=True
```

Values: on=`True`, off=`False`

If “next word/ char” or “prev word/char” cannot be performed because you reached the end of the screen, announce it:

```
endOfScreen=True
```

Values: on=`True`, off=`False`

Leave the review mode when pressing a key:

```
leaveReviewOnKeypress=False
```

Values: on=`True`, off=`False`

Leave the review mode when changing the screen (From TTY3 to TTY4):

```
leaveReviewOnScreenChange=True
```

Values: on=`True`, off=`False`

Promote

“Promoted Lists” are configured in the section `[promote]`. Turn Promoted Lists“ on or off:

```
enabled=True
```

Values: on=`True`, off=`False`

The minimum time interval of inactivity to activate promoting. By default it promotes after 120 Seconds inactivity:

```
inactiveTimeoutSec=120
```

Values: in Seconds

Define a list of promoted words comma seperated:

```
list=
```

Values: text (comma seperated) Example to promote the word “nickname” or a bash prompt:

```
list=nickname,$:,#:
```

Time

The automated time announcement is configured in the section `[time]`. Time announcement is disabled by default. Turn time announcement on or off:

```
enabled=True
```

Values: on=`True`, off=`False`

Should the time be announced:

```
presentTime=True
```

Values: on=`True`, off=`False`

Should the date be announced (just on date change):

```
presentDate=True
```

Values: on=`True`, off=`False`

Announce after a given period of seconds:

```
delaySec=0
```

Value: in Seconds, 0 = Deactivated

Announce after fixed minutes in an hour. if delaySec is >0 onMinutes is ignored:

```
onMinutes=00,30
```

Example every 15 minutes:

```
onMinutes=00,15,30,45
```

Just play a soundicon, (not interrupting):

```
announce=True
```

Values: on=`True`, off=`False`

Interrupt current speech for time announcement:

```
interrupt=False
```

Values: on=`True`, off=`False`

Customization

Scripting

Scripts can be in any language, bash, python, sh or others. Place your scripts in the directory /usr/share/fenrirscreenreader/scripts/ (the path is configurable in settings.conf). The script key is the applications key. Usually this key can be found on the keyboard located just left of the right most control key. When you name a script, the key name appears in the script seperated by the sequence - . So, for example, if you have a python weather script you want assigned to the script key plus the letter w you would name the script /usr/share/fenrirscreenreader/scripts/weather-key_w.py Then, to access the script, simply press the script key and the letter w. Scripts must be executable. So, make sure to chmod 755 your script when you place it in the scripts directory. The script gets some parameters from fenrir when it is executed. So that information is available in your script then.

Parameterlist

Parameter	Content
$1	Username of the current logged in user

Examples

Script that just speaks the current username when pressing ScriptKey + H.
File: `/usr/share/fenrirscreenreader/scripts/helloWorld-key_h.sh`:

```
#!/bin/bash
echo $1
```

Commands

You can place your own commands in ”/usr/share/fenrirscreenreader/commands“ (path is configurable in settings.conf). Commands are python files with a special scheme. You can assign them to a shortcut using the filename without an extension or place them in a hook trigger like OnInput or OnScreenChange. For further information see developer guide. Good Examples: "date.py" (announce the Date), "shut_up.py" (interrupt output) the basic scheme for a command is as follows:

```
from core import debug

class command():
    def __init__(self):
        pass
    def initialize(self, environment):
        self.env = environment
    def shutdown(self):
        pass
    def getDescription(self):
        return _('No description found')
    def run(self):
        pass
    def setCallback(self, callback):
        pass
```

- Template lives here
- The class needs to have the name “command”.
- “initialize” is running once whilst loading the command.
- “shutdown” is running on unload like the command (quit fenrir)
- “getDescriptsion” just returns an string. That String is used in Tutorial Mode.
- “run” is executed when the command is invoked. (shortcut is pressed, or trigger isn't running)
- setCAllback is currently not used. and has no functionality yet.

Troubleshooting

Below are fenric instructions that do not all apply to Slint.

Slint users may request help on the Slint mailing list.

Fenrir does not start

1. Have you installed all the dependencies Support and Requirements
2. Try using master, a lot of changes take place there to make Fenrir compatible with more systems

Fenrir does not utilize the shortcuts

1. Make sure you have python3-evdev installed
2. Use the latest Fenrir version
3. Make sure that Fenrir has permission to /dev/input/* and /dev/uinput (or run it as root)

No sound at all

1. Run the script to configure Pulseaudio once as root and once as your user. This will setup Pulseaudio but require a restart of Pulseaudio. The script is located in `tools/configure_pulse.sh`

2. Use ALSA
3. Configure Pulse system wide (Not recommended)
4. Use gstreamerDriver: change `settings.conf` in the section `sound` the line `driver=genericDriver` to `driver=gstreamerDriver`
5. Use wave sound-icons: change `settings.conf` in the section `sound` the line `theme=default` to `theme=default-wav`
6. Use most current version of sox with opus support
7. Try apulse (not tested by myself but might work). Please give me feedback if you try it out.

You get sound-icons but no speech

1. If you are using speech-dispatcher run “spd-conf” once as user and as root.
2. You can test if speech-dispatcher works by invoking it as root
   ```
   sudo spd-say “hello world”
   ```

Bugreports and feature requests

Please report Bugs and feature requests to: https://github.com/chrys87/fenrir/issues

for bugs please provide a debug file that shows the issue.

How-to create a debug file

1. Delete old debug stuff
   ```
   sudo rm /var/log/fenrir.log
   ```
2. Start fenrir in debug mode
   ```
   sudo fenrir -d
   ```
3. Do your stuff to reproduce the problem
4. Stop fenrir (`fenrirKey + q`)

the debug file is located in `/var/log/fenrir.log`

Please be as precise as possible to make it easy to solve the problem.

From:
/wiki/ - **Slint**

Permanent link:
/wiki/doku.php?id=playground:fenrir&rev=1587467111

Last update: **2020/04/21 11:05**