

Emacspeak – The Complete Audio Desktop

User Manual

T. V. Raman

Copyright © 1994–2008 T. V. Raman. All Rights Reserved.

Permission is granted to make and distribute verbatim copies of this manual without charge provided the copyright notice and this permission notice are preserved on all copies.

Short Contents

Emacspeak	1
1 Copyright	2
2 Announcing Emacspeak Manual 2nd Edition As An Open Source Project	3
3 Introduction	4
4 Installation Instructions	5
5 Basic Usage	8
6 The Emacspeak Audio Desktop	18
7 Voice Lock	21
8 Using Online Help With Emacspeak	23
9 Emacs Packages	25
10 Running Terminal Based Applications	45
11 Emacspeak Commands	49
12 Emacspeak Customizations	134
13 Emacspeak TTS Servers	196
14 Acknowledgements	199
15 Concept Index	200
16 Key Index	201

Table of Contents

Emacspeak	1
1 Copyright	2
2 Announcing Emacspeak Manual 2nd Edition As An Open Source Project	3
2.1 How To Contribute To This Manual	3
2.2 Authoring Guidelines	3
2.3 Credits	3
3 Introduction	4
4 Installation Instructions	5
4.1 Obtaining Emacspeak	5
4.2 Quick Installation	5
4.3 Configuring and Installing Emacspeak	6
5 Basic Usage.	8
5.1 Overview of Emacspeak	8
5.2 Working In Emacs Buffers	9
5.3 Reading Without Moving The Cursor	9
5.4 Speech System Commands	12
5.4.1 Character, Word And Line Echo	12
5.4.2 Setting Various Characteristics Of Speech Output	12
5.4.3 Miscellaneous Speech Commands	14
5.5 Voice Lock Mode	16
5.6 Commands For Speaking Status Information	16
6 The Emacspeak Audio Desktop.	18
6.1 Objects Making Up The Emacspeak Desktop	18
6.2 An Object-Oriented Desktop	18
6.3 Emacspeak Specializes Aural Interaction	19
6.3.1 Audio Formatted Output	19
6.3.2 Structured Navigation:	19
6.3.3 Navigating The Desktop	19
6.3.4 Everything Is Searchable:	19
7 Voice Lock	21
7.1 How It All Works	21
7.2 What this gives is	22

8	Using Online Help With Emacspeak.	23
9	Emacs Packages.	25
9.1	Document Authoring.	25
9.1.1	Creating Well-formatted Documents.	25
9.1.2	Searching, Replacing, And Spell Checking	26
	Spell Checking	27
9.2	Structured Editing And Templates.	28
9.2.1	Outline Editing	28
9.2.2	Template-based Authoring	29
9.2.3	Maintaining Structured Data.	29
9.3	Browsing Structured Information.	30
	Related Packages.	30
9.5	Web Browsing.	30
9.5.1	Intro.	30
9.5.2	emacs-w3m	31
9.5.3	emacs/w3.	34
9.5.4	Add-ons.	39
9.5.5	Conclusion	40
9.6	Electronic Messaging Applications.	41
	Related Packages.	41
9.8	Editing Program Source Code.	41
	Related Packages.	41
9.10	Software Development Environment	41
	Related Packages.	41
9.12	Desktop Management	41
	Related Packages.	42
9.14	Personal Information Management	42
	Related Packages.	42
9.16	Desktop Applications	42
9.16.1	Spread Sheets.	42
9.16.2	Forms Mode	42
9.16.3	OCR –Reading Print Documents	42
9.16.3.1	Emacspeak OCR Mode.	43
	Related Packages.	44
10	Running Terminal Based Applications.	45
10.1	Char Sub-mode of Term Mode	45
10.2	Line Sub-mode of Term Mode.	47
10.3	Eterm Mode Commands	48

11	Emacspeak Commands	49
11.1	cd-tool	49
11.2	dtk-speak	49
11.3	emacspeak-alsaplayer	52
11.4	emacspeak-amphetadesk	55
11.5	emacspeak-arc	55
11.6	emacspeak-aumix	55
11.7	emacspeak-bs	56
11.8	emacspeak-buff-menu	56
11.9	emacspeak-c	56
11.10	emacspeak-calendar	57
11.11	emacspeak-compile	57
11.12	emacspeak-custom	57
11.13	emacspeak-daisy	57
11.14	emacspeak-dired	59
11.15	emacspeak-dismal	59
11.16	emacspeak-ecb	60
11.17	emacspeak-ediff	61
11.18	emacspeak-enriched	61
11.19	emacspeak-entertain	61
11.20	emacspeak-eperiodic	61
11.21	emacspeak-erc	62
11.22	emacspeak-eterm	63
11.23	emacspeak-filtertext	66
11.24	emacspeak-fix-interactive	66
11.25	emacspeak-forms	67
11.26	emacspeak-gnus	67
11.27	emacspeak-gomoku	67
11.28	emacspeak-gridtext	68
11.29	emacspeak-hide	69
11.30	emacspeak-ibuffer	69
11.31	emacspeak-imenu	69
11.32	emacspeak-info	70
11.33	emacspeak-jabber	70
11.34	emacspeak-keymap	70
11.35	emacspeak-loaddefs	70
11.36	emacspeak-m-player	71
11.37	emacspeak-madplay	74
11.38	emacspeak-man	74
11.39	emacspeak-ocr	74
11.40	emacspeak-outline	77
11.41	emacspeak-pronounce	77
11.42	emacspeak-python	79
11.43	emacspeak-realaudio	79
11.44	emacspeak-redefine	80
11.45	emacspeak-remote	81
11.46	emacspeak-rmail	82
11.47	emacspeak-solitaire	82

11.48	emacspeak-sounds	82
11.49	emacspeak-speak	83
11.50	emacspeak-speedbar	93
11.51	emacspeak-table-ui	93
11.52	emacspeak-tabulate	96
11.53	emacspeak-tapestry	96
11.54	emacspeak-tar	96
11.55	emacspeak-tetris	97
11.56	emacspeak-tnt	97
11.57	emacspeak-url-template	98
11.58	emacspeak-view-process	98
11.59	emacspeak-view	98
11.60	emacspeak-vm	99
11.61	emacspeak-w3	99
11.62	emacspeak-w3m	101
11.63	emacspeak-we	101
11.64	emacspeak-websearch	104
11.65	emacspeak-webutils	109
11.66	emacspeak-widget	110
11.67	emacspeak-wizards	110
11.68	emacspeak-xml-shell	121
11.69	emacspeak-xslt	121
11.70	emacspeak-zinf	122
11.71	emacspeak	122
11.72	voice-setup	123
11.73	emacspeak-moz	123
11.74	emacspeak-org	125
11.75	emacspeak-webmarks	125
11.76	URL Templates	126

12 Emacspeak Customizations **134**

12.1	cd-tool Options	134
12.2	dtk-speak Options	134
12.3	emacspeak-add-log Options	135
12.4	emacspeak-advice Options	136
12.5	emacspeak-alsaplayer Options	136
12.6	emacspeak-amphetadesk Options	137
12.7	emacspeak-aumix Options	137
12.8	emacspeak-calendar Options	137
12.9	emacspeak-compile Options	137
12.10	emacspeak-custom Options	138
12.11	emacspeak-daisy Options	138
12.12	emacspeak-dired Options	139
12.13	emacspeak-dismal Options	139
12.14	emacspeak-ediff Options	139
12.15	emacspeak-entertain Options	139
12.16	emacspeak-eperic Options	140
12.17	emacspeak-erc Options	140

12.18	emacspeak-eshell Options	140
12.19	emacspeak-eterm Options	142
12.20	emacspeak-forms Options	143
12.21	emacspeak-gnus Options	143
12.22	emacspeak-hide Options	143
12.23	emacspeak-ido Options	143
12.24	emacspeak-imenu Options	144
12.25	emacspeak-info Options	144
12.26	emacspeak-ispell Options	144
12.27	emacspeak-jabber Options	144
12.28	emacspeak-keymap Options	146
12.29	emacspeak-loaddefs Options	147
12.30	emacspeak-m-player Options	148
12.31	emacspeak-madplay Options	148
12.32	emacspeak-make-mode Options	148
12.33	emacspeak-message Options	149
12.34	emacspeak-muse Options	149
12.35	emacspeak-ocr Options	150
12.36	emacspeak-outline Options	151
12.37	emacspeak-pcl-cvs Options	151
12.38	emacspeak-personality Options	152
12.39	emacspeak-pronounce Options	152
12.40	emacspeak-re-builder Options	153
12.41	emacspeak-realaudio Options	154
12.42	emacspeak-remote Options	154
12.43	emacspeak-replace Options	154
12.44	emacspeak-rpm-spec Options	154
12.45	emacspeak-sigbegone Options	155
12.46	emacspeak-sounds Options	155
12.47	emacspeak-speak Options	156
12.48	emacspeak-speedbar Options	158
12.49	emacspeak-table-ui Options	158
12.50	emacspeak-tnt Options	158
12.51	emacspeak-url-template Options	159
12.52	emacspeak-vm Options	159
12.53	emacspeak-w3 Options	160
12.54	emacspeak-w3m Options	160
12.55	emacspeak-we Options	161
12.56	emacspeak-websearch Options	161
12.57	emacspeak-webutils Options	161
12.58	emacspeak-wizards Options	162
12.59	emacspeak-xml-shell Options	164
12.60	emacspeak-xslt Options	164
12.61	emacspeak Options	165
12.62	voice-setup Options	165

13	Emacspeak TTS Servers	196
13.0.1	High-level Overview	196
13.0.2	Commands That Queue Output	196
13.0.3	Commands That Set State	197
14	Acknowledgements	199
15	Concept Index	200
16	Key Index	201

Emacspeak

This manual documents Emacspeak, a speech interface to Emacs.

1 Copyright

This manual documents Emacspeak, a speech extension to Emacs.

Copyright (C)1994 – 2008 T. V. Raman All Rights Reserved.

Permission is granted to make and distribute verbatim copies of this manual without charge provided the copyright notice and this permission notice are preserved on all copies.

2 Announcing Emacspeak Manual 2nd Edition As An Open Source Project

This is to announce the launch of a new open source project to create a user manual for Emacspeak –an Emacs speech extension that provides a complete audio desktop.

2.1 How To Contribute To This Manual

This manual is organized as a series of chapters, with each chapter in a separate file. If you feel capable of contributing to a specific section, send out a message to the Emacspeak mailing list `emacspeak@cs.vassar.edu`. You can then start adding content to a local copy of the chapter to which you are contributing. When you feel you have something to submit, mail out the file to the emacspeak mailing list– I’ll integrate new content as it comes in.

2.2 Authoring Guidelines

For this manual to hang together and make sense to the new user at whom it is targeted, contributors need to stick to a consistent style. If you plan to contribute content, you should take some time to read the existing sections –note that many of these are skeletal and the first contributions will be to flesh these sections out.

If you are familiar with texinfo, go ahead and mark up your content using texinfo. If you are not, simply author the documentation you create as plain formatted ASCII. If you do submit files as texinfo source, make sure to validate them at your end first by running the files through `makeinfo` –badly created or malformed texinfo source takes more time to fix than marking up straight text.

2.3 Credits

This initial version draws heavily from the original Emacspeak user manual, and includes contributions from Jim Van Zandt and Jason White. Authors who contribute complete sections will be acknowledged here as well as in the specific section they author.

3 Introduction

Emacspeak provides a complete audio desktop by speech-enabling all of Emacs.

In the past, screen reading programs have allowed visually impaired users to get feedback using synthesized speech. Such programs have been commercially available for well over a decade. Most of them run on PC's under DOS, and these are now moving over to the Windows environment. However, screen-readers for the UNIX environment have been conspicuous in their absence. Note that this is now changing with the availability of console-level Linux screenreaders such as `speakup`. Such Linux screenreaders provide the same level of UNIX accessibility provided in the late 80's by PC terminal emulators running a DOS screenreader. This means that most visually impaired computer users face the additional handicap of being DOS-impaired — a far more serious problem:-)

Emacspeak is an emacs subsystem that provides complete speech access. It is *not* a screen-reader —rather, it is a complete user environment with built-in speech feedback. Emacspeak has a significant advantage; since it runs inside Emacs, a structure-sensitive, fully customizable environment, Emacspeak has more context-specific information about what it is speaking than its screenreader counterparts. This is why Emacspeak is not a “screenreader”, it is a subsystem that produces speech output.

A Traditional screen-reader speaks the content of the screen, leaving it to the user to interpret the visual layout. Emacspeak, on the other hand, treats speech as a first-class output modality; it speaks the information in a manner that is easy to comprehend when listening.

The basic concepts used by Emacspeak are simple; all interactive Emacs commands have been adapted to provide speech feedback. Hence, you use Emacs as normal; Emacspeak works behind the scene to give audio feedback in addition to updating the screen.

Emacspeak consists of a core speech system that provides speech and audio services to the rest of the Emacspeak desktop; application-specific extensions provide context-specific spoken feedback using these services. Emacspeak currently comes with speech extensions for several popular Emacs subsystems and editing modes. I would like to thank their respective authors for their wonderful work which makes Emacs more than a text editor¹.

¹ I have now been using Emacspeak under Linux as the only source of speech feedback since 1994.

4 Installation Instructions

This chapter gives brief and detailed installation instructions for configuring, installing and starting Emacspeak.

4.1 Obtaining Emacspeak

Emacspeak is available on the Internet at:

WWW <http://emacspeak.sf.net>

WWW <http://www.cs.cornell.edu/home/raman/emacspeak/>

FTP <ftp://ftp.cs.cornell.edu/pub/raman/emacspeak>

Mail List emacspeak@cs.vassar.edu

List Request
 emacspeak-request@cs.vassar.edu

The Emacspeak mailing list is maintained by Greg E. Priest-Dorman. If you are using Emacspeak, you can join the list by sending mail to the request address.

4.2 Quick Installation

Here are the quick installation instructions. See the next section for detailed installation instructions.

Prepackaged RPM files are available on the Emacspeak site. Packages for other Linux distributions such as Debian typically become available on the WWW a few days after a new version is released. The instructions below are for building and installing Emacspeak from the source distribution.

- Type ‘`make config`’ to configure the sources for your site.
- Type ‘`make`’ to compile the files.
- Install the system by typing ‘`make install`’
- Run it by doing one of:
 - ‘`emacspeak`’
 - Add the line

```
(load-file "<emacspeak-dir>/lisp/emacspeak-setup.el")
```

to the top of your `.emacs` file.
-

In the above, `<emacspeak-dir>` refers to the directory where you unpacked the sources.

4.3 Configuring and Installing Emacspeak

Note: You need a current version of GNU Emacs for using newer versions of Emacspeak.

The speech server for the Dectalk is written in TclX. (For example, see the source file ‘`dtk-exp`’).

- Decide where you want the Emacspeak elisp files. Most Unix packages are unpacked and compiled in one directory, and then the executable, documentation, and data files are installed in the appropriate system directories.

Once you have decided on the appropriate directory, make it if necessary and unpack the distribution `.tar` file there.

- Look through the introductory section of the Makefile to see if you need to edit it for your site.
- Check the directory prefix for installation. The default is ‘`/usr/local`’, so ‘`.info`’ files go to ‘`/usr/local/info`’ and the executable file goes to ‘`/usr/local/bin`’. To change this, edit the line defining *PREFIX*.

Configure the source files by typing ‘`make config`’. At this point you can check that the speech server is correctly configured by typing

```
tcl dtk-exp
```

(assuming you are using the Dectalk Express). You should hear the Dectalk speak and get a TCL prompt if everything is okay.

If you’re feeling paranoid, you can perform a couple of additional tests at this point. Execute the following commands in the running tcl session you just started above. (Most users will not need to do this; it is a sanity check and is useful in tracking problems, especially if you find emacspeak beginning to talk and then immediately fall silent.)

- ```
q "this is a test."
d
```

You should hear the Dectalk speak the text.

- ‘`s`’ The above command stops speech. You should see a TCL prompt when you execute it. If things appear to hang when you execute ‘`s`’ i.e. you don’t see a TCL prompt
  1. The serial cable connecting your speech device is flaky
  2. Your serial port is flaky
  3. The stty settings on the port are incorrect for your system

In the case of incorrect stty settings please report the problem. The following UNIX command can be used to report the stty settings:

```
stty -a < serial_port
```

Quit this TCL session by typing `C-D`.

Next, compile the elisp files by typing

```
make emacspeak
```

Finally, install the documentation and executable files by typing

```
make PREFIX=<prefix> install
```

The speech server program and/or output port can also be specified at run time by setting the shell environment variables *DTK\_PROGRAM* and *DTK\_PORT*. Examples: If using *cs*h or *tc*sh

```
setenv DTK_PROGRAM "dtk-exp"
```

if using *sh* or *bash*

```
DTK_PROGRAM=dtk-exp
export DTK_PROGRAM
```

Similarly,

```
DTK_PORT=/dev/ttyS0
```

You can always set these variables from a running Emacs session by executing the Emacs *setenv* command.

## 5 Basic Usage.

This chapter gives an overview of how to use Emacspeak. Note: This documentation should be used in conjunction with the online Emacs info pages that extensively document Emacs itself. These sections briefly describe the speech-enabling extensions. However, they should not be considered a substitute for reading the Emacs manual. How successfully you use Emacspeak will depend on how well you learn your Emacs.

All Emacs navigation and editing commands have been speech enabled. Thus, moving to the next or previous word, line or paragraph results in the text around point being spoken. Exactly how much text is spoken is determined by the amount by which you moved.

In addition, Emacspeak provides basic reading functions that can be invoked to listen to chunks of text without moving.

### 5.1 Overview of Emacspeak

Emacspeak provides a small number of core services around which the remainder of the audio interface is constructed. These essential features of the software are briefly outlined in the following paragraphs; the commands by which they can be controlled will be described later in the manual.

Apart from providing a fluent spoken interface to all of Emacs' basic editing functions, Emacspeak also includes software modules which add speech feedback to a range of applications that can be run from within Emacs. In this sense, Emacspeak amounts to much more than a talking text editor; indeed, it can more aptly be characterized as a true “audio desktop”, in which speech is treated as a first-class output modality.

Emacspeak implements a special minor mode, known as “voice lock mode” (see Section 5.5 [Voice-lock], page 16) which uses distinct speech characteristics to provide aural highlighting of specific textual constructs, such as comments in program code, quoted strings and reserved words See [\[Voice Lock\]](#), page [\[undefined\]](#). This facility is further extended when Emacspeak is used with the W3 World Wide Web browser, to enable the semantic and structural distinctions captured by the HTML markup to be communicated efficiently See Section 9.5 [Web Browsing], page 30.

It is often desirable to exercise control over the pronunciation of a word (E.G. a technical term or a reserved word in a programming language) within specific contexts. Emacspeak maintains pronunciation dictionaries for this purpose, which may be customized by the user. Moreover, individual dictionaries can be activated selectively, depending for example on the current major mode or the name of the file which is being visited See Section 11.41 [emacspeak-pronounce], page 77.

In addition to spoken feedback, Emacspeak can generate “auditory icons”—short sound cues which alert the user to significant events, for example the opening or deletion of a file, the completion of an action, the arrival of an electronic mail message or the creation of a completion buffer. Sound cues act as a supplement to the spoken interface, and are especially valuable to the experienced user in facilitating rapid interaction. Note that in order to support auditory icons, the computer must be equipped with sound hardware for which the operating system has been correctly configured See Section 11.48 [emacspeak-sounds], page 82.

## 5.2 Working In Emacs Buffers.

While typing in an Emacs buffer, hitting space speaks the recently typed word. I use completion all the time; so Emacspeak will speak the completion just inserted as well as the next possible completion. In Emacs, use `load-library ret completion ret` for loading the completion package.

The standard Emacs prompting functions have also been speech-enabled. Emacs prompts with available lists of completions in response to partial input wherever appropriate—all forms of completion provide speech feedback.

In addition, Emacspeak provides a number of commands for reading portions of the current buffer, getting status information, and modifying Emacspeak's state.

All of the commands are documented in the subsequent sections. They can be classified into types:

- Emacspeak commands for listening to chunks of information. The names of these commands all start with the common prefix `emacspeak-speak-`. All Emacspeak commands are bound to the keymap `emacspeak-keymap` and are accessed with the key `Control e`<sup>1</sup>. Thus, the Emacspeak command `emacspeak-speak-line` is bound to `l` in keymap `emacspeak-keymap` and can be accessed with the keystroke `Control-e l`. If for some reason you wish to use some key other than `control-e` as the common keyboard prefix for all Emacspeak commands, set the variable `emacspeak-prefix`.

- 

The second category of commands provided by Emacspeak manipulate the state of the speech device. The names of these commands start with the common prefix `dtk-` and are bound in keymap `emacspeak-dtk-submap`. You can access these commands via the prefix `Control-e d`<sup>2</sup>. Thus, the command `dtk-set-rate` is bound to `r` in keymap `emacspeak-dtk-submap` and can be executed by pressing `Control e d r`.

Emacs has extensive online help; so does emacspeak. Please use it.

This info manual is only to get you started. You can get a summary of Emacspeak's features by pressing `Control-h Control-e`

## 5.3 Reading Without Moving The Cursor.

Emacspeak speaks information as you move around within a buffer. How much text is spoken depends on how you move, thus, when you move by words, you hear the current word; when you move by paragraphs, you hear the current paragraph spoken. In addition, the following commands allow you to listen to information without moving point (point is emacs terminology for the editing cursor).

Reading without moving point:

`control e c`

emacspeak-speak-char Speak character under point. Pronounces character phonetically unless called with a PREFIX arg.

---

<sup>1</sup> Control e is mnemonic for Emacspeak.

<sup>2</sup> d is mnemonic for Dectalk.

**control e w**

emacspeak-speak-word Speak current word. With prefix ARG, speaks the rest of the word from point. Negative prefix arg speaks from start of word to point. If executed on the same buffer position a second time, the word is spelled instead of being spoken.

**control e l**

emacspeak-speak-line Speaks current line. With prefix ARG, speaks the rest of the line from point. Negative prefix optional arg speaks from start of line to point. Voicifies if option ‘voice-lock-mode’ is on. Indicates indentation with a tone if audio indentation is in use. Indicates position of point with an aural highlight if option ‘emacspeak-show-point’ is turned on –see command ‘emacspeak-show-point’ bound to M-x emacspeak-show-point. Lines that start hidden blocks of text, e.g. outline header lines, or header lines of blocks created by command ‘emacspeak-hide-or-expose-block’ are indicated with auditory icon ellipses.

**control e up**

emacspeak-read-previous-line Read previous line, specified by an offset, without moving. Default is to read the previous line.

**control e down**

emacspeak-read-next-line Read next line, specified by an offset, without moving. Default is to read the next line.

**control e {**

emacspeak-speak-paragraph Speak paragraph. With prefix arg, speaks rest of current paragraph. Negative prefix arg will read from start of current paragraph to point. If voice-lock-mode is on, then it will use any defined personal-ity.

**control e r**

emacspeak-speak-region Speak current region delimited by *point* and *mark*. When called from a program, argument START and END specify region to speak.

**control e cap R**

emacspeak-speak-rectangle Speak a rectangle of text. Rectangle is delimited by point and mark. When call from a program, arguments specify the START and END of the rectangle.

**control e b**

emacspeak-speak-buffer Speak current buffer contents. With prefix ARG, speaks the rest of the buffer from point. Negative prefix arg speaks from start of buffer to point. If voice lock mode is on, the paragraphs in the buffer are voice annotated first, see command ‘emacspeak-speak-voice-annotate-paragraphs’.

**control e n**

emacspeak-speak-rest-of-buffer Speak remainder of the buffer starting at point

*control e /*

emacspeak-speak-this-buffer-other-window-display Speak this buffer as displayed in a different frame. Emacs allows you to display the same buffer in multiple windows or frames. These different windows can display different portions of the buffer. This is equivalent to leaving a book open at places at once. This command allows you to listen to the places where you have left the book open. The number used to invoke this command specifies which of the displays you wish to speak. Typically you will have two or at most three such displays open. The current display is 0, the next is 1, and so on. Optional argument ARG specifies the display to speak.

*control e left*

emacspeak-speak-this-buffer-previous-display Speak this buffer as displayed in a ‘previous’ window. See documentation for command ‘emacspeak-speak-this-buffer-other-window-display’ for the meaning of ‘previous’.

*control e right*

emacspeak-speak-this-buffer-next-display Speak this buffer as displayed in a ‘previous’ window. See documentation for command ‘emacspeak-speak-this-buffer-other-window-display’ for the meaning of ‘previous’.

*control e [*

emacspeak-speak-page Speak a page. With prefix ARG, speaks rest of current page. Negative prefix arg will read from start of current page to point. If option ‘voice-lock-mode’ is on, then it will use any defined personality.

*control e 9 control e 8 control e 7 control e 6 control e 5 control e 4 control e 3 control e 2 control e 1 control e 0*

emacspeak-speak-predefined-window Speak one of the first 10 windows on the screen. In general, you’ll never have Emacs split the screen into more than two or three. Argument ARG determines the ‘other’ window to speak. Speaks entire window irrespective of point. Semantics of ‘other’ is the same as for the builtin Emacs command ‘other-window’.

*control e control n*

emacspeak-speak-next-window Speak the next window.

*control e control p*

emacspeak-speak-previous-window Speak the previous window.

*control e control o*

emacspeak-speak-other-window Speak contents of ‘other’ window. Speaks entire window irrespective of point. Semantics of ‘other’ is the same as for the builtin Emacs command ‘other-window’. Optional argument ARG specifies ‘other’ window to speak.

*ESCAPE up*

emacspeak-owindow-previous-line Move to the next line in the other window and speak it. Numeric prefix arg COUNT specifies number of lines to move.

*ESCAPE down*

emacspeak-owindow-next-line Move to the next line in the other window and speak it. Numeric prefix arg COUNT can specify number of lines to move.

**ESCAPE next**

emacspeak-owindow-scroll-up Scroll up the window that command ‘other-window’ would move to. Speak the window contents after scrolling.

**ESCAPE prior**

emacspeak-owindow-scroll-down Scroll down the window that command ‘other-window’ would move to. Speak the window contents after scrolling.

**control e ’**

emacspeak-speak-sexp Speak current sexp. With prefix ARG, speaks the rest of the sexp from point. Negative prefix arg speaks from start of sexp to point. If option ‘voice-lock-mode’ is on, then uses the personality.

**control e meta control @**

emacspeak-speak-spaces-at-point Speak the white space at point.

## 5.4 Speech System Commands

This section documents Emacspeak’s various user commands for controlling the text to speech (TTS) system.

### 5.4.1 Character, Word And Line Echo.

By default, Emacspeak speaks characters as they are typed –this is called character echo; Words are spoken as they are completed –this is called word echo. Emacspeak can also optionally speak each line as it is typed –this is called line echo.

Character, word and line echo can be toggled –either in the current buffer– or for all buffers (globally). To toggle the specific echo functionality for all buffers, precede the specific command with *C-u*. Note that in the documentation below, this use of *C-u* is indicated using the common Emacs terminology of *prefix arg* or *interactive prefix arg*.

**control e d k**

emacspeak-toggle-character-echo Toggle state of Emacspeak character echo. Interactive PREFIX arg means toggle the global default value, and then set the current local value to the result.

**control e d w**

emacspeak-toggle-word-echo Toggle state of Emacspeak word echo. Interactive PREFIX arg means toggle the global default value, and then set the current local value to the result.

**control e d l**

emacspeak-toggle-line-echo Toggle state of Emacspeak line echo. Interactive PREFIX arg means toggle the global default value, and then set the current local value to the result.

### 5.4.2 Setting Various Characteristics Of Speech Output.

Emacspeak user commands can set different characteristics of the speech output such as speech rate and punctuations mode.

Emacspeak provides a number of settings that affect how attributes of the text such as capitalization are conveyed. These include settings that produce a short tone for each

upper case letter, as well as a smart mode for speaking mixed case words which is especially useful when programming. These settings can be made locally in a given buffer or be applied to all buffers by preceding these commands with *C-u*.

*control e d r*

*dtk-set-rate* Set speaking RATE for the tts. Interactive PREFIX arg means set the global default value, and then set the current local value to the result.

*control e d f*

*dtk-set-character-scale* Set scale FACTOR for speech rate. Speech rate is scaled by this factor when speaking characters. Interactive PREFIX arg means set the global default value, and then set the current local value to the result.

This function is advised.

Before-advice ‘*emacspeak-auto*’: Automatically defined advice to speak interactive prompts.

*control e d 9 control e d 8 control e d 7 control e d 6 control e d 5*

*control e d 4 control e d 3 control e d 2 control e d 1 control e d 0*

*dtk-set-predefined-speech-rate* Set speech rate to one of nine predefined levels. Interactive PREFIX arg says to set the rate globally.

*control e d p*

*dtk-set-punctuations* Set punctuation mode to MODE. Possible values are ‘some’, ‘all’, or ‘none’. Interactive PREFIX arg means set the global default value, and then set the current local value to the result.

*control e d m*

*dtk-set-pronunciation-mode* Set pronunciation MODE. This command is valid only for newer Dectalks, e.g. the Dectalk Express. Possible values are ‘math, name, europe, spell’, all of which can be turned on or off. Argument STATE specifies new state.

*control e d s*

*dtk-toggle-split-caps* Toggle split caps mode. Split caps mode is useful when reading Hungarian notation in program source code. Interactive PREFIX arg means toggle the global default value, and then set the current local value to the result.

*control e d c*

*dtk-toggle-capitalization* Toggle capitalization. when set, capitalization is indicated by a short beep. Interactive PREFIX arg means toggle the global default value, and then set the current local value to the result.

*control e d cap C*

*dtk-toggle-allcaps-beep* Toggle allcaps-beep. when set, allcaps words are indicated by a short beep. Interactive PREFIX arg means toggle the global default value, and then set the current local value to the result. Note that allcaps-beep is a very useful thing when programming. However it is irritating to have it on when reading documents.

In addition, Emacspeak can convey the indentation of lines as they are spoken— this is relevant when programming and is the default when working with program source.

*control e d i*

emacspeak-toggle-audio-indentation Toggle state of Emacspeak audio indentation. Interactive PREFIX arg means toggle the global default value, and then set the current local value to the result. Specifying the method of indentation as ‘tones’ results in the Dectalk producing a tone whose length is a function of the line’s indentation. Specifying ‘speak’ results in the number of initial spaces being spoken.

Indentation feedback style is set by option emacspeak-audio-indentation-method

The default value is "speak"

See variable ‘emacspeak-audio-indentation-methods’ for possible values. Automatically becomes local in any buffer where it is set.

### 5.4.3 Miscellaneous Speech Commands

Speech can be stopped using command `dtk-stop`—though in normal use, the action of moving the cursor will stop ongoing speech. Speech can also be paused and resumed. The speech server can be stopped and restarted for cases where the user wants to switch to a different server—or in the rare case to nuke a runaway speech server.

*Control e s*

`dtk-stop` Stop speech now.

*control e p*

`dtk-pause` Pause ongoing speech. The speech can be resumed with command ‘`dtk-resume`’ normally bound to C-e SPC. Pausing speech is useful when one needs to perform a few actions before continuing to read a large document. Emacspeak gives you speech feedback as usual once speech has been paused. ‘`dtk-resume`’ continues the interrupted speech irrespective of the buffer in which it is executed. Optional PREFIX arg flushes any previously paused speech.

*control e SPACE*

`dtk-resume` Resume paused speech. This command resumes speech that has been suspended by executing command ‘`dtk-pause`’ bound to C-e p. If speech has not been paused, and variable ‘`dtk-resume-should-toggle`’ is t then this command will pause ongoing speech.

*control e d q*

`dtk-toggle-quiet` Toggle state of the speech device between being quiet and talkative. Useful if you want to continue using an Emacs session that has emacspeak loaded but wish to make the speech shut up. Optional argument PREFIX specifies whether speech is turned off in the current buffer or in all buffers.

*control e control s*

*dtk-emergency-restart* Use this to nuke the currently running dtk server and restart it. Useful if you want to switch to another synthesizer while emacspeak is running. Also useful for emergency stopping of speech.

Finally, here are the remaining commands available via the TTS related keymap *C-e d*.

*control e d a*

*dtk-add-cleanup-pattern* Add this pattern to the list of repeating patterns that are cleaned up. Optional interactive prefix *arg* deletes this pattern if previously added. Cleaning up repeated patterns results in emacspeak speaking the pattern followed by a repeat count instead of speaking all the characters making up the pattern. Thus, by adding the repeating pattern ‘.’ (this is already added by default) emacspeak will say “aw fifteen dot” when speaking the string “.....” instead of “period period period period”.

*control e d d*

*dtk-select-server* Select a speech server interactively. This will be the server that is used when you next call either *M-x dtk-initialize* or *C-e C-s*. Argument *PROGRAM* specifies the speech server program.

*control e d SPACE*

*dtk-toggle-splitting-on-white-space* Toggle splitting of speech on white space. This affects the internal state of emacspeak that decides if we split text purely by clause boundaries, or also include whitespace. By default, emacspeak sends a clause at a time to the speech device. This produces fluent speech for normal use. However in modes such as ‘shell-mode’ and some programming language modes, clause markers appear infrequently, and this can result in large amounts of text being sent to the speech device at once, making the system unresponsive when asked to stop talking. Splitting on white space makes emacspeak’s stop command responsive. However, when splitting on white space, the speech sounds choppy since the synthesizer is getting a word at a time.

*control e d RETURN*

*dtk-set-chunk-separator-syntax* Interactively set how text is split in chunks. See the Emacs documentation on syntax tables for details on how characters are classified into various syntactic classes. Argument *S* specifies the syntax class.

*control e d t*

*emacspeak-dial-dtk* Prompt for and dial a phone *NUMBER* with the Dectalk.

*control e d cap V*

*emacspeak-dtk-speak-version* Use this to find out which version of the TTS firmware you are running.

*control e d z*

*emacspeak-zap-dtk* Send this command to the TTS engine directly.

## 5.5 Voice Lock Mode

The status of voice lock mode can be toggled on and off by issuing the command `C-e d v` (`M-x voice-lock-mode`). With a prefix arg, this function applies globally; otherwise, it is local to the current buffer. To have voice lock mode activated automatically when Emacspeak starts, include the following code in your `.emacs` file:

```
(add-hook 'emacs-lisp-mode-hook
 'turn-on-voice-lock)
```

Alternatively, Emacspeak can be set to enable voice lock automatically in all of the major modes that support it. To do so, insert the following statement into your Emacs initialization file:

```
(global-voice-lock-mode t)
```

Note that the list of major modes in which `global-voice-lock-mode` will provide automatic activation is specified in the variable `voice-lock-global-modes`. See Chapter 7 [Voice Lock], page 21.

The characteristics of the different voice personalities deployed by voice lock mode vary according to the capabilities of the speech synthesizer. The definitions applicable to the Dectalk family of synthesizers are contained in `'dtk-voices.el'`, which is supplied as part of the Emacspeak distribution.

Using voice lock mode, Emacspeak also supports many of the aural style properties defined in level 2 of the World Wide Web Consortium's Cascading Style Sheet specification (see <http://www.w3.org/TR/REC-CSS2/>). Thus, when Emacspeak is running in conjunction with a cooperating user agent, such as William Perry's Emacspeak/W3 web browser, the rendering of HTML documents can be regulated by style sheets. Examples of style rules which employ the CSS audio properties can be found in the default style sheet which is supplied in the Emacs/W3 distribution.

## 5.6 Commands For Speaking Status Information.

The following commands provide miscellaneous information.

*control e a*

emacspeak-speak-message-again Speak the last message from Emacs once again.

*control e m*

emacspeak-speak-mode-line Speak the mode-line.

*control e cap M*

emacspeak-speak-minor-mode-line Speak the minor mode-information.

*control e control w*

emacspeak-speak-window-information Speaks information about current windows.

*control e t*

emacspeak-speak-time Speak the time.

*control e cap V*

emacspeak-speak-version Announce version information for running emacspeak.

- control e f*  
emacspeak-speak-buffer-filename Speak name of file being visited in current buffer. Speak default directory if invoked in a dired buffer, or when the buffer is not visiting any file.
- control e h*  
emacspeak-speak-help Speak help buffer if one present. With prefix arg, speaks the rest of the buffer from point. Negative prefix arg speaks from start of buffer to point.
- control e k*  
emacspeak-speak-current-kill Speak the current kill entry. This is the text that will be yanked in by the next C-y. Prefix numeric arg, COUNT, specifies that the text that will be yanked as a result of a C-y followed by count-1 M-y be spoken. The kill number that is spoken says what numeric prefix arg to give to command **yank**.
- control e v*  
emacspeak-view-register Display the contents of a register, and then speak it.
- control e control @*  
emacspeak-speak-current-mark Speak the line containing the mark. With no argument, speaks the line containing the mark—this is where ‘exchange-point-and-mark’ C-x C-x would jump. Numeric prefix arg ‘COUNT’ speaks line containing mark ‘n’ where ‘n’ is one less than the number of times one has to jump using ‘set-mark-command’ to get to this marked position. The location of the mark is indicated by an aural highlight achieved by a change in voice personality.
- control e control l*  
emacspeak-speak-line-number Speak the line number of the current line.
- control e =*  
emacspeak-speak-current-column Speak the current column.
- control e %*  
emacspeak-speak-current-percentage Announce the percentage into the current buffer.

## 6 The Emacspeak Audio Desktop.

This chapter describes the Emacspeak audio desktop and gives tips and tricks for making use of many of Emacs' powerful features.

The desktop is the work area where you organize the tools of your trade and the information objects relevant to your current activities. In the conventional world of visual GUI-based computing, these tools and information objects manifest themselves as a collection of icons organized in a two-dimensional work-area –this organization is designed to place frequently used objects within easy reach.

Notice that organizing one's work area in terms of visual icons arranged in a two-dimensional area where such an organization is optimized for the available "conversational gestures" of pointing and clicking is an artifact of visual interaction.

In the spirit of a truly speech-enabled application, Emacspeak does not simply provide you spoken access to a particular presentation of your work environment that was initially designed with the "sign language" of visual interaction in mind. Instead, Emacspeak enables you to work with documents and other information objects in a manner that is optimized to aural, eyes-free interaction. A necessary consequence of this setup is that users accustomed to the purely visual manifestation of today's electronic desktop do not immediately perceive the Emacspeak environment as an electronic desktop. This section of the manual hopes to introduce you to a work-style that encourages a different perspective on how one interacts with the computer in performing day-to-day computing tasks.

The end result in my case has been a marked increase in personal productivity.

### 6.1 Objects Making Up The Emacspeak Desktop

A "buffer" is the basic building block of the Emacs and hence the Emacspeak desktop. Any information presented by Emacs is placed in a "buffer". For example, when perusing this manual within Emacs, the "file" containing the documentation is presented in a "buffer". All information objects such as WWW pages, email messages, output from user interaction with command-line shells etc., are presented by Emacs in individual "buffers".

Buffers provide a base level of user interaction; Emacs derives its power by allowing applications to specialize buffers to enable specific types of user-interaction that is optimized for a specific class of information.

### 6.2 An Object-Oriented Desktop

The basic "buffer object" can be specialized by Emacs applications to provide optimal interaction. This kind of specialization makes the Emacs environment an object-oriented environment; thus, the basic conversational gesture of "move to the next statement" can be assigned behavior that is appropriate to the content that the user is currently navigating. As an example of such specialization, Emacs provides "specialized modes" for working with English text, programming languages, markup source e.g. HTML or LaTeX documents and so on.

## 6.3 Emacspeak Specializes Aural Interaction

The content-specific user interaction described above is a very powerful feature of Emacs, and this is where Emacspeak derives its power. Traditionally, the ability to create buffers specialized for working with specific content-types has been used by the Emacs community to develop versatile programming environments, messaging applications such as mail and news readers, and authoring environments. The clean design present in all of these Emacs extensions in terms of separating application functionality from the user-interface, combined with the availability of the entire source code making up these packages under the open-source model has laid the ground-work for developing Emacspeak as a versatile aural counterpart to the product of years of software engineering that has been invested by the Emacs community. In short, Emacspeak would not exist in its present shape or form without this prior effort.

### 6.3.1 Audio Formatted Output

Emacspeak takes advantage of the content-specific knowledge available within specialized buffers to produce "audio formatted" output designed to optimize user interaction. A basic consequence of the above is "voice locking" in specialized modes; a more interesting consequence is the implementation of Aural Cascading Style Sheets (ACSS) in conjunction with the Emacs W3 browser.

### 6.3.2 Structured Navigation:

Emacspeak also exploits content-specific knowledge to provide structured navigation of different types of electronic content. In many cases, such structured navigation is an extension of what Emacs provides by default; in other cases, Emacspeak implements the necessary extensions to provide the level of structural navigation needed to work efficiently in an eyes-free environment.

Notable among such structured navigation is Emacs' powerful outline feature. Notice for example, that the Emacspeak FAQ (reached via command `emacspeak-view-emacspeak-faq` bound to `C-e F`) takes advantage of Emacs' outline mode to allow you to easily move through the various sections. An example of content-sensitive navigation is provided by the `imenu` package which dynamically creates a "table of contents" based on the content that is being displayed in a given buffer.

### 6.3.3 Navigating The Desktop

In addition to navigating individual information objects, the Emacspeak environment provides speech-enabled navigation of the various buffers that are currently open on the Emacspeak desktop via Emacs' built-in list-buffers feature. Emacs' `dired`—directory editor— for browsing the file system, along with the new `speedbar` package that combines features from `dired` and `imenu` round off the suite of navigational tools.

### 6.3.4 Everything Is Searchable:

Emacs derives one final advantage from using buffers as the basic building block for the entire desktop. Every Emacs buffer is searchable via a uniform and powerful search interface. Emacs' incremental search works efficiently and consistently to enable you locate "objects" of interest either within a given document or to locate a given object from among the various objects that are currently open on the Emacspeak desktop. This is

**very powerful** –where a GUI user is typically limited to quickly locating an object from a relatively small collection –the size of the collection being a direct function of available display real-estate– the Emacspeak user can typically work with a far larger collection of objects. This is well-suited to the eyes-free environment, where display real-estate has no meaning; so bringing up a list of currently open buffers and performing an incremental search to locate a specific buffer is just as efficient independent of whether you have a few dozen or a few hundred buffers open.

To illustrate the above, my typical working Emacs session lasts between two and three weeks– over that time I typically accumulate several hundred open buffers holding a large variety of content ranging from program source code to email messages and WWW pages.

Ubiquitous search in the eyes-free environment is critical– as a comparison, when using a conventional, purely visual WWW browser, users have no means of easily "searching" for say the "submit" button on a WWW page. This inability is a minor annoyance in visual interaction, and the typical mouse-enabled user **never** uses the find dialog to find a submit button– it is simply more efficient to point at the submit button given the eye's ability to quickly scan the two-dimensional display. This luxury is absent in an eyes-free environment; as a consequence, blind users confronted by the combination of a visual interface and screen-reader are typically limited to either tabbing through all the controls on a WWW page, or using the sub-optimal find dialog.

## 7 Voice Lock

1. Emacspeak defines a number of voice overlays such as ‘`voice-bolden`’, and ‘`voice-lighten`’ that can be applied to a given voice to change what it sounds like.
2. Voice overlays are defined in terms of Aural CSS (ACSS (<http://www.w3.org/tr/css2/aural.html>) to keep them independent of a specific TTS engine.
3. For each such overlay there is a corresponding ‘`<overlay-name>-settings`’ variable that can be customized via custom.
4. The numbers in ‘`voice-bolden-settings` as an example’:

| Setting       | Value |
|---------------|-------|
| family        | nil   |
| average-pitch | 1     |
| pitch-range   | 6     |
| stress        | 6     |
| richness      | nil   |
| punctuation   | nil   |

Unset values (‘`nil`’) show up as “unspecified” in the customize interface.

1. Do not directly customize ‘`voice-bolden`’ and friends, instead customize the corresponding ‘`voice-bolden-settings`’, since that ensures that all voices that are defined in terms of ‘`voice-bolden`’ get correctly updated.
2. Discovering what to customize:

Command ‘`emacspeak-show-personality-at-point`’ (bound by default to C-e M-v) will show you the value of properties personality and face at point. A recent update I implemented last weekend makes this more useful, so make sure you do a CVS update; earlier this command used to display the ACSS setting — now it displays the abstract name. Describe-variable on these names should tell you what to customize; so as an example:

Put point on a comment line, and hit ‘C-e M-v’: you will hear

```
Personality emacspeak-voice-lock-comment-personality
Face font-lock-comment-delimiter-face
```

Describe-variable of ‘`emacspeak-voice-lock-comment-personality`’ gives:

```
emacspeak-voice-lock-comment-personality's value is acss-p0-s0-all
```

Documentation:

```
Personality used for font-lock-comment-face
```

```
This personality uses voice-monotone whose effect can be changed globally by customizing
```

### 7.1 How It All Works

Here is a brief explanation of the connection between ‘`voice-bolden`’ and its associated ‘`voice-bolden-settings`’.

1. Voice settings are initially in ‘`voice-bolden-settings`’ which is a list of numbers.
2. That list of numbers needs to be translated to appropriate device-specific codes to send to the TTS engine.

3. You do not want to do this translation *each* time you speak something.
4. So when ‘`voice-bolden`’ is defined, the definition happens in two steps:
  - The list of settings is stored away in ‘`voice-bolden-settings`’,
  - A corresponding voice-name is generated — ‘`acss-a<n>-p<n>-r<n>-s<n>`’ and the corresponding control codes to send to the device are stored away in a hash-table keyed by the above symbol.
  - Finally, ‘`voice-bolden`’ is assigned the above symbol.

## 7.2 What this gives is

1. The ability to customize the voice via custom by editing the list of numbers in ‘`voice-bolden-settings`’
2. When that list is edited, ‘`voice-bolden`’ is arranged to be updated automatically.

The following additional commands from module See Section 11.67 [emacspeak-wizards], page 110, are useful when designing aural styles.

1. `emacspeak-wizards-generate-voice-sampler`  
Generate a buffer containing text that demonstrates the effect of various aural settings.
2. `emacspeak-wizards-voice-sampler`  
Applied specified aural style to text in current region.

## 8 Using Online Help With Emacspeak.

Emacs provides an extensive online help system for helping you learn about various aspects of using Emacs. Emacspeak provides online help for its various extensions using this same help system. This chapter explains how to use the online help facilities in order to empower you in discovering powerful and versatile working techniques that will make you more and more productive in your day to day computing.

The online help options are accessed via the `C-h` prefix key, which must be followed by an additional letter or control character to designate the kind of help desired. For example, `C-h t help-with-tutorial` visits the Emacs tutorial in a new buffer; `C-h i info` enters the Info documentation system, from which you can read Texinfo manuals that have been installed on your system, including the Emacs and Emacspeak documentation; and `C-h k describe-key` provides a description of the Emacs function which is bound to the next key that you type. For learning about the various options that are available via the `C-h` mechanism described above, view the online help for command `help-for-help` bound to `C-h C-h`—using what has been described so far, you would achieve this by pressing `C-h k` followed by `C-h C-h`.

Emacspeak users should note that online help is typically displayed in a separate Emacs window. Where it makes sense to do so, Emacspeak will automatically speak the displayed help. Once you've asked for help, you can have the displayed documentation as many times as you wish using Emacspeak command `emacspeak-speak-help` bound to `C-e h`. If you want to move through the displayed help a line at a time, switch to the buffer where the help is displayed—the buffer is called `*Help*`.

Often, in adding an auditory interface to an Emacs extension, such as a web browser or mail reader, Emacspeak defines additional commands and key bindings which enhance the functionality of the spoken feedback provided by the application. This manual does not purport to document all such commands. It is important, therefore, when learning to use the various Emacs extensions which comprise the 'audio desktop' (see Chapter 6 [Audio Desktop], page 18) that you take advantage of online help to obtain details of any context-specific features provided by Emacspeak. The following two commands are of particular importance in this regard:

- `C-h m describe-mode` explains which major and minor modes are currently in effect, and lists the commands and key bindings associated with them.
- `C-h b describe-bindings` lists all of the key bindings which are currently defined.

The importance of these help functions can be illustrated by the Emacs/W3 web browser. When point is positioned inside a table, certain key bindings are established with which you can access Emacspeak commands that make it possible to read the rows and columns of the table and explore its structure efficiently. To get a description of these key bindings, you can use W3 to visit the sample HTML file supplied as part of the Emacspeak distribution, and, after having moved point onto the first row of the table, issue the command `C-h m describe-mode` to create a help buffer containing an explanation of the features offered by W3 mode.

Emacspeak supplements the online help facilities available within Emacs by defining several commands of its own, as follows:

- `C-h c-e describe-emacspeak` presents a list of standard Emacspeak commands.

- *C-e F* `emacspeak-view-emacspeak-faq` opens a new buffer containing the Emacspeak FAQ, a list of frequently asked questions about Emacspeak together with their answers.
- *C-e C-h* `emacspeak-learn-mode` enters a mode in which the function of every key that you type is spoken; this mode can be terminated with the *C-g* `keyboard-quit` command.
- *M-x* `emacspeak-generate-documentation` prompts for the name of a file in which it saves an automatically generated summary of all Emacspeak commands.

## 9 Emacs Packages.

Emacs – The extensible, self-documenting editor, derives its functionality from its powerful extension mechanism. This extension mechanism is used to implement many user-level applications such as mail readers, WWW browsers, software development environments and so on. This chapter gives directions on how to locate the right Emacs package for addressing specific tasks. The chapter is organized into logical sections that each pertain to a specific class of tasks; in individual subsections within a section give a brief overview of particular Emacs packages that have been speech-enabled.

### 9.1 Document Authoring

The Emacspeak environment provides a rich collection of structured document authoring tools. These are well-suited for working in an eyes-free environment –you clearly do not want to use a **What You See Is What You Get** (WYSIWYG) authoring tool if you cannot see what you're getting. Structure-based authoring tools allow you to focus on the act of content creation, leaving the minutiae of visual layout to the computer.

#### 9.1.1 Creating Well-formatted Documents

Before authoring a document, decide its primary audience if the document contains relatively simple content e.g., no mathematical equations etc. and is primarily targeted at the WWW, you are probably better off using HTML. You can create well-structured HTML documents with the help of package `html-helper-mode` available from `ftp://ftp.reed.edu/pub/src/html-helper-mode.tar.gz` — note this is now mostly obsolete by James Clarke's excellent `nxml-mode` for editing XML documents.

Package `html-helper-mode` is speech-enabled by Emacspeak to provide auditory icons, structured navigation and outlines, as well as voice locking for audio formatted feedback as you work.

If the document being authored is more complex, you are usually better off creating it in LaTeX. Note that LaTeX documents can be converted to HTML either via package `tex4ht` – available on the WWW.

The TeX family of typesetting languages is suitable for producing well-formatted documents in an eyes-free environment. Unlike WYSIWYG environments, the author of a TeX or LaTeX document works with the content of the document, leaving it to the formatting system (TeX) to format the document for good visual presentation.

The `auctex` package is an Emacs extension that facilitates authoring and maintaining structured documents in TeX and LaTeX. Package `bibtex` facilitates maintenance and use of `bibtex` bibliography databases. The `Texinfo` package allows creation of software documentation that is suitable for both printing as well as online viewing as hypertext. Emacspeak speech-enables packages `auctex`, `bibtex` and `texinfo` to provide convenient spoken feedback as you create and compile documents. For details on using these packages, see their accompanying online info documentation.

The most recent version of package `auctex` is always available by ftp at `ftp://ftp.iesd.auc.dk/pub/emacs-lisp/auctex.tar.gz`. Packages `bibtex` and `texinfo` are part of the standard Emacs distribution.

As the document preparation system of choice, Emacspeak supports a fluent speech-enabled interface to editing and formatting LaTeX documents. This interface is provided by speech-enabling *auctex* mode.

Mode *auctex* provides efficient keyboard shortcuts for inserting and maintaining LaTeX markup as a document is being authored. All of these editing commands provide succinct auditory feedback when used with Emacspeak. The syntax coloring provided by this mode is extended to provide *voice locking* — consequently, Emacspeak uses different voices to speak the embedded markup to set it apart from the content.

Mode *auctex* can be used to create empty document templates and to insert document content at the appropriate places in the template. The mode also enables structured navigation of the document as it is being edited. Emacspeak speech-enables these template creation and structured navigation commands to produce auditory icons and succinct spoken feedback. For example, while editing, the user can quickly browse through the sections of the document and have each section title spoken. Document elements such as paragraphs and bulleted lists can be manipulated as logical units. These features are especially relevant in an eyes-free environment where the user needs to select logical parts of the document without having to point at portions of a visual display.

Finally, Emacs supports creating and maintaining SGML and XML documents. Emacs comes with a fairly simple `sgml-mode` — in addition, package `psgml` provides sophisticated parsing and validation facilities for working with SGML and XML documents. Package `psgml` can be downloaded from [http://w4.lns.cornell.edu/public/COMP/info/psgml/psgml\\_toc.html](http://w4.lns.cornell.edu/public/COMP/info/psgml/psgml_toc.html).

### 9.1.2 Searching, Replacing, And Spell Checking

Incremental search, a process by which the system prompts the user for a search string and moves the selection to the next available match while allowing the user to add more characters to the search string, is the search technique of choice among most Emacs users. As the system successively finds each match and provides the user the option of continuing the search. Incremental search is a more complex instance of traditional search interaction because in addition to either stopping or continuing the search, the user can modify the current search in a number of ways including specifying a longer (or shorter) search string.

All of the user commands available during incremental search are documented in the online Emacs info manual. These are speech-enabled by Emacspeak to provide spoken prompts as the dialogue begins; auditory icons indicate a search hit or search miss as the search progresses. Along with auditory icons *search-hit* and *search-miss* the user also hears the current line spoken, and in the case of a search hit, the matching text is *aurally* highlighted by using the standard audio formatting technique of changing voice characteristic. This feedback proves extremely effective when the search pattern appears several times on a single line; the user is unambiguously cued to the current match.

Search and replace actions are an extension to the basic conversational gestures of a search dialogue. In addition to specifying a search string, the user also specifies a replacement string. On the Emacspeak desktop, this functionality is provided by command *query-replace*. The speech-enabled version of this interaction prompts the user for the search and replacement texts. The auditory feedback during the interactive search and

replacement process parallels that described in the case of incremental search. Audio formatting to indicate the occurrence that is about to be replaced proves an effective means of avoiding erroneous modifications to the text being edited. As an example, consider using command *query-replace* to locate and replace the second occurrence of **foo** with **bar** in the text

```
'Do not change this fool, but change this food.'
```

When the search matches the first occurrence of **foo** in word *fool*, the aural highlighting helps the user in answering “no” in response to question “should this occurrence be replaced”. In addition to allowing the user to supply a simple “yes or no” answer for each match, command *query-replace* also allows the user to specify a number of other valid answers as described in the online Emacs documentation.

## Spell Checking

A more complex instance of conversational gesture “search and replace” is exhibited by standard spell checking dialogues. Spell checking differs from the search and replace dialogue described above in that the search and replacement text is guessed by the system based on an available dictionary. Words that are not found in the dictionary are flagged as potential spelling errors, and the system offers an interactive search and replace dialogue for each of these possible errors. During this dialogue, the system successively selects each occurrence of the possibly erroneous word and offers a set of possible replacements. Unlike in the case of simple search and replace, more than one possible replacement string is offered, since a potential spelling error can be corrected by more than one word appearing in the dictionary.

In the visual interface, such spell checking dialogues are realized by displaying the available choices in a pop-up window and allowing the user to pick a correction. Once a correction is selected, the user is offered the choice of interactively replacing the erroneous word with the correction.

The spell checking interface on the Emacspeak desktop is speech-enabled to provide fluent auditory feedback. The visual interface parallels that described above and is provided by package `ispell` which is part of the standard Emacs distribution. Emacspeak provides a spoken prompt that is composed of the line containing the possibly erroneous word (which is aurally highlighted to set it apart from the rest of the text on that line) and the available corrections. Each correction is prefixed with a number that the user can use to select it. Once a correction is selected, the interaction continues with the query and replace interaction described earlier. The speech interface to the spell checker is as fluent as the visual interface. Notice that Emacspeak users do not need to concern themselves with the details of the visual display such as “the corrections are displayed in a window at the top of the screen”.

In addition to the standard spell checker described above, newer versions of Emacs include an “on-the-fly” spell checker that flags erroneous words as they are typed. Emacspeak speech-enables package `flyspell` so that such erroneous words are aurally highlighted.

## 9.2 Structured Editing And Templates

Editing documents based on the inherent structure present in the electronic encoding can be very efficient when using spoken interaction. We described mode *auctex* — a specialized interface to authoring LaTeX documents as a special instance of such structured editing in see Section 9.1 [Document Authoring], page 25.

The Emacspeak desktop allows the user to efficiently author and maintain an electronic document based either on the structure present in the markup (as in the case of mode *auctex*) or on special outlining constructs that allow the user to impose a desired logical structure on the document. This section describes the effect of speech-enabling such editing tools and points out the advantages in using these in a speech oriented interface.

Template-based authoring — a technique that allows the user to create a document by inserting contents into appropriate positions in a predefined template— goes hand in hand with such structured editing. Finally, structured editing can vastly simplify the creation and maintenance of structured data, for example, the data present in a UNIX password file. Such data files are in fact nothing more than a collection of database records, where each record (or line) consists of a set of fields delimited by a special character. Maintaining such files without exploiting the underlying structure often tends to be error prone. We describe editing modes that can exploit such record structure to provide a fluent editing interface. Finally, we outline a speech-enabled interface to a spreadsheet application as a complex instance of such structured data editing.

### 9.2.1 Outline Editing

All of the various outline editing interfaces on the Emacs desktop allow the user to *hide* or *show* the contents at the different levels of a possibly nested tree structure. Components of this tree structure can be manipulated as a unit, e.g., entire subtrees can be deleted or copied. Outline editing thus provides an efficient means of obtaining quick overviews of a document.

The visual interface displays such hidden content as a series of ellipses following the visible outline heading. Emacspeak produces auditory icon *ellipses* when speaking such outline headings.

The basic *outline mode* allows the user to specify the syntax and level of outline header lines as a regular expression. This simple technique can be used to advantage in the structured navigation of large electronic texts such as those available on the Internet from online book projects such as project Gutenberg and the Internet Wiretap. For example, when this feature is activated while reading the electronic text of a Shakespearean play, the different acts can be recognized as separate nodes in the logical structure of the document. The user can then hide the document body with a single keystroke, navigate the outline headings to find a particular act, and have that portion rendered either visually or aurally. Hiding an outline level produces auditory icon *close-object*; exposing a hidden level produces auditory icon *open-object*. For details on using mode `outline`, see the relevant section of the online Emacs info manual.

The basic outline facility described above is applicable to all content being edited or browsed on the Emacspeak desktop. In addition, Emacspeak has other specialized outline editing modes such as *folding mode* that provide extended outlining facilities. In

mode *folding*, the user can create (possibly nested) *folds* — logical containers of content that are delimited by a special *fold mark*. The fold mark is typically a text string that is chosen based on the type of content that is being manipulated. Thus, when *folding* a C~program source file, fold marks are created from C~comments. The user can *open* or *close* any or all folds in a document, and these actions are accompanied by auditory icons *open-object* and *close-object*. By entering a fold, all editing actions are restricted to the contents of that fold; this proves a simple yet convenient way of constraining editing actions such as search and replace to specific portions of large documents. Folds can be manipulated as a unit and can be deleted, copied or moved.

Mode *folding* proves especially effective in maintaining large software modules. The technique can be used to advantage by creating folds for different sections in a module and by further placing each function appearing in a particular % section in a fold of its own. Complex functions can themselves be folded into sections where each section reflects a different stage in the algorithm implemented by that function. Thus, the technique of folding can be used as an effective aid in *literate* programming. I typically write software modules by first creating an outline structure using folds that reflect the various components of that module. Next, I populate each fold with the function signatures and documentation for the functions in each section. When I am satisfied with the overall architecture of the module, I fill in the function skeletons with actual program code. This technique is used extensively in maintaining the Emacspeak code base.

### 9.2.2 Template-based Authoring

Emacspeak supports two powerful template-based authoring subsystems that enable the user to quickly create and fill in templates. *Dmacro* (short for “dynamic macros”) allows the user to define and invoke template-based macros that are specialized for creating different types of content. For example, when programming in C, the user can invoke dynamic macros that insert skeletons of standard C constructs with a few keystrokes. This form of editing has numerous advantages in creating consistently structured code when developing large software modules. Emacspeak speech-enables mode *dmacro* to provide succinct spoken feedback as templates are created and filled. The user invokes *dmacro* via command *insert dmacro*, which is typically bound to a single key. This results in a dialogue where the user is prompted to pick one of the dynamic macros available in the current context. If the users choice can be uniquely completed, that completion is spoken; otherwise, the list of possible completions based on the available partial input is spoken, accompanied by auditory icon *help*.

An alternative template-editing facility is provided by mode *tempo*. This mode is designed to be used in creating template-based editing tools for specific markup languages; a good example is mode *html-helper*, a mode for creating and updating HTML documents for publishing on the WWW (see see Section 9.1 [Document Authoring], page 25).

### 9.2.3 Maintaining Structured Data

Consider the following entry from file `/etc/passwd` on my laptop.

```
'aster:KoUxwQ2:501:100:Aster Labrador:/home/aster:/bin/bash'
```

File `/etc/passwd` is a simple instance of a text file that stores structured data records as a series of fields delimited by a special character. Each item in the file acquires *meaning* from the *position* in which it occurs for example, the fifth field contains the

**user name**, Aster Labrador. More generally, structured data where each field in a record has *meaning* is found throughout the desktop in applications ranging from entries in a rolodex to rows in a spreadsheet.

Typically, users do not directly edit the stored representation of the data. Instead, application front-ends provide a more human-centric (and hopefully less error prone) user interface for modifying and maintaining the data. Thus, spreadsheet applications present the data as a two dimensional table that is automatically updated to reflect changes in the underlying data. The two dimensional table is perhaps the most commonly found visual front-end to structured data tables with row and column headers prove a succinct way of implicitly displaying the *meaning* along with the *value* of the fields making up each data record.

## 9.3 Browsing Structured Information

This section describes packages that allow you to browse structured information –these are distinct from the tools described in Section 9.2 [Structured Editing], page 28, in that they are typically used for working with content that is read-only e.g., online documentation.

## Related Packages

### 9.5 Web Browsing.

This document, “The State of Web Browsing in Emacspeak” describes the primary web browsers in use under emacspeak and ways they might be used more efficiently.

Version 1.0 – February, 2007.

Copyright © 2007 Robert D. Crawford `rdc1x@comcast.net`

#### 9.5.1 Intro

Web browsing in emacspeak can be made as simple or as complicated as you wish to make it. On the one hand, all that need be done is to open a web browser and fetch a particular url. This works fine in many cases with exceptions being things like html tables. On the other hand, if you spend a little time learning a particular browser and its functionality, web browsing can be made more efficient and the web more navigable.

This document is intended to be an introduction to the 2 primary browsers that run under emacs: emacs/w3 and emacs-w3m. I also intend to introduce a couple of add-on packages intended to make life easier and to address specific shortcomings.

The primary reason for this documentation is the fact that a lot of the documentation of the various functions tells exactly what it does, but not why or in what circumstance one might use it. Hopefully I can remedy that here.

### Who this document is intended for

While this document is geared toward emacspeak users, it might be helpful for other users as well. I have tried to note where functions are specific to emacspeak.

## Assumptions

This document assumes emacs is installed. If reading the section on a particular browser and trying out the associated functions, it, of course, assumes that the browser is installed and working. For features specific to emacspeak, a working install of emacspeak is necessary. It is far beyond the scope of this document to help with the installation of these programs.

For help with installing any of the above bits of software, the following mailing lists and / or newsgroups are available:

- emacs - `gnu.emacs.help`
- emacs-w3m - see `http://emacs-w3m.namazu.org`
- emacs/w3 - the mailing list for w3 is pretty dead. Sorry. The documentation is, however, not terrible.
- emacspeak - see `http://emacspeak.sf.net`

It is highly suggested that the mailing list archives, google, and the relevant documentation be consulted before posting messages to any mailing list. Nothing is more irritating than answering the same questions over and over. Those of you with kids know what I am talking about.

It is also assumed that the reader is comfortable with using emacs itself. Understanding the convention for communicating keystrokes to run commands, navigating documents, and the like are not covered.

### 9.5.2 emacs-w3m

emacs-w3m is a front-end, written in emacs lisp, to the text-based web browser w3m. It supports tables, images, and all major web protocols. It is also fully supported in emacspeak.

#### emacs-w3m advantages

The main advantage of emacs-w3m is speed. The rendering of web pages is done with the external program w3m. which is written in c, so it is fast. It also seems not to choke on some web pages like the alternative, emacs/w3.

Another big advantage to emacs-w3m is that it is under active development. Problems seem to be fixed soon after they come to the attention of the developers and are available very quickly as long as you are willing to use the cvs version.

For those who have limited sight as opposed to no sight at all, emacs-w3m displays all graphics within its own buffer. This might be a misconfiguration on my part, but there are some images that open in an external window when opened from emacs/w3.

I can also say that I cannot remember a page that wouldn't render in emacs-w3m. This is not something I can say about emacs/w3.

#### emacs-w3m disadvantages

Taken by itself, the disadvantages of using w3m are few. Long periods between releases can be seen by some to be a disadvantage although the alternative browser's time between releases is now several years, I believe.

With hardware or software that support multiple voices, such as IBM's Via Voice, one can allow differences in speech to indicate differences in document text. For example, headings can be spoken in a voice with a lower pitch, links in a higher voice. The voices in emacs-w3m are much more limited than those in emacs/w3 and not all textual attributes can be indicated. As of the writing of this document, `pre` tags and the like cannot be indicated. It has been within the last few months that italic text support has been added.

Another disadvantage is the fact that the rendering in emacs-w3m is done by the external process, w3m. Since w3m is written in C, it is not as easily customizable from within the community of emacs users. Code written in emacs lisp can be substituted for code in a running emacs instance, something that is not possible in C.

A further disadvantage is the lack of table navigation, i.e. the ability to move from cell to cell in a table to more easily understand the information presented. Linearizing the table is available but sometimes it is difficult to keep the meanings of the columns in your head to discern the meaning. Maybe it is just me.

A disadvantage shared between the browsers is the lack of support for javascript. Many sites use javascript for links to the next page, such as Reuters web site. This can sometimes be worked around using the google transcoder. See See [below], page 33, for more information.

## emacs-w3m native functions

This part of the document is not meant to replace the `describe-mode` information available via the `C-h m` keystroke or the emacs-w3m info manual. This is by no means an exhaustive list of available commands.

emacs-w3m has considerable functionality and many features that make browsing more efficient and convenient. Below are many of the functions that are native to emacs-w3m.

`Cap S` will query for a search term. By default, the search engine used is google. This can be changed with a prefix command. To see the available engines, after hitting `C-u Cap S`, when it asks for the engine hit `TAB` to get a list of available completions. Tab completion of partial words works here as well. For example, hitting "y" and then `TAB` will complete to the word "yahoo".

Form navigation can be achieved with the right and left brackets. This is useful for times where you want to go to the input field on a web page without having to tab through the links. Google is a good example of where this would be useful. The right bracket moves point to the next form, and the left bracket moves to the previous, wrapping around at the ends of the buffer.

While in a form, using `C-c C-c` will submit the form with no need to tab around looking for the "Submit" button.

## emacspeak specific functions for emacs-w3m

Many extensions have been written by the emacspeak community to expand emacs-w3m and make it more accessible. Below are those extensions.

While being a variable and not a function, the variable `emacspeak-w3m-speak-titles-on-switch` makes emacs-w3m more friendly. It tells emacspeak to speak the document title when switching between emacs-w3m buffers. The default, `nil`, will speak the mode line. This is the same behavior seen in all other buffers. This speaking of the document title is more descriptive than just hearing `*w3m*<4>`. This variable can be customized in the usual ways, either in your `.emacs` file or via the `Customize` interface. If you do set this variable, reading the mode line can still be achieved with `C-e m`.

The `j` command will jump to the first occurrence of the title in the document. Each time you use this command in succession, it will jump to the next occurrence of the title. This is very convenient for web pages that have the title show up in multiple places in the document. The web site for The Register is a good example of this.

The `l` command uses the default media player to play the media stream at point. I believe this is set to `emacspeak-m-player` by default. While beyond the scope of this document to describe the installation of mplayer, I would highly advise using this media player. With the correct codecs installed, mplayer will play everything you will find, I believe, with one convenient player that does not require a GUI.

`C-t` will run the current page through the google transcoder. The transcoder is very useful to the users of screen readers as it takes care of formatting issues like tables. While not usually a problem, there is a function to linearize tables in emacs-w3m, but it is not always successful. You can reverse this command by giving a prefix argument.

`M-t` runs the url under point through the transcoder. This command can be reversed by giving a prefix argument.

Extensions have been written to take advantage of much of the google functionality. The following describes how to access many of these extensions.

`C-c C-g` will search the current web site with google.

`C-c C-x` will extract the current page from the google cache. A prefix argument will extract the url under point from the cache. This is useful if, for instance, a site is down for one reason or another and you absolutely must have the information.

`C-c C-l` will find pages similar to the current page. This is just like the “similar” link at the end of a google search result.

`C-c C-r` will browse the rss link under point. One reason this might be useful is to first try out the rss feed before you configure your rss reader to subscribe to the feed. For a easy to use rss reader, see the section below on See [`emacspeak-rss`], page 40.

The letter `x` is the prefix key for the xsl transforms. `xslt` is some powerful ju-ju. It takes a web page and does all kinds of cool things with it. These things are listed below.

The command `x a` will apply an xsl transformation to the current page. one particularly useful transform is to linearize tables. This takes the tables and formats them linearly down the page as if they were paragraphs instead of table cells. This command can be more easily applied with the `x l` command.

If you want to always have tables linearized, use the `x a` command and type `linearize TAB` and then the `x o` to toggle xsl on. This will linearize the tables in every subsequently visited page for this session. While it seems possible to make this happen automatically in emacs/w3, I do not believe this is currently possible in emacs-w3m. A noticeable increase in speed can be seen in emacs/w3 by turning `xslt` on and using `sort-tables` or `linearize-tables`. This eliminates the need for the browser to render nested

tables. Since speed is not such an issue with emacs-w3m, I don't bother turning on xslt for the session.

The `x b` command will add a “submit” button to forms that do not have one such as the one on the emacs wiki web site. Submitting forms can be more easily achieved with `C-c C-c`.

If on a google search page, the `x h` command will give you only the search hits. An easier way of getting the same results is to use the `url-template` “Google Hits” which returns the same results. Another advantage of using the `url-template` is that it can be done from anywhere, not just in an emacs-w3m buffer. See [emacspeak-url-template], page 39, for more information.

The `x t` command sorts the tables. It takes the nested tables and unnests them. While sometimes useful, this is not nearly as useful as the same function in emacs/w3. While doing the same thing, emacs/w3 has the capability of allowing the user to navigate the tables cell by cell.

## emacs-w3m tips and tricks

One issue that might need work is that, by default, when tabbing over links in emacs-w3m the url of the link is spoken. There are several ways to deal with this.

One solution, actually two but they accomplish the same result, is to include the following code in the `~/ .emacs-w3m` file:

```
(remove-hook 'w3m-after-cursor-move-hook
 #'w3m-print-this-url)
```

Or, this code in the `.emacs` file

```
(add-hook 'w3m-after-cursor-move-hook
 (lambda ()
 (emacspeak-speak-messages nil)))
```

Both of these solutions, as far as the user can see, seem to be equivalent. They differ only in where they are placed.

The other solution is to leave things as they are. Some users mention that they find it helpful to have the url spoken and, if they want the text of the link spoken they use `C-e`. As with everything else in emacs, it is completely up to you to decide what is best.

### 9.5.3 emacs/w3

emacs/w3 is a web browser written completely in emacs lisp. It has some really nice features applicable to the emacspeak community such as the ability to navigate tables and support for the w3c's aural cascading stylesheets.

## emacs/w3 advantages

As mentioned above, the ability to navigate tables is a super help. emacs/w3 also has support for cascading stylesheets. This allows incredible control of voices used for what would normally be visual attributes of the text such as bold, italics, preformatted text and the like.

Another advantage of emacs/w3 is that it is written completely in emacs lisp. With some effort, emacs/w3 is very customizable and quite extendable.

## emacs/w3 disadvantages

Rendering can be slow. Sometimes it can be painfully, excruciatingly slow. That might be a slight exaggeration, but slow it is. This is because it is written in lisp... something I mentioned just above as a strength. It is a trade-off, but one that some see as worth it. There are, however, some things that can be done to speed up the browsing process. See See [emacspeak-w3-xsl-transform], page 38, for more information.

Another disadvantage is that emacs/w3 chokes on some pages. Sometimes it gives error messages and doesn't display anything. Sometimes it does this to some people and doesn't do it to others as we saw on the emacspeak mailing list a short time ago. Sometimes it gives error messages and renders the page anyway.

As mentioned in the emacs-w3m section, this browser does not support javascript, which is a problem, usually for links on some sites. One way this problem can be sometimes worked around is to transcode the link under point via google. See See [w3 transcode], page 37, for more information.

One other major annoyance of emacs/w3 is that sometimes it simply stops doing anything while rendering a page. There is a way I have found to get around this. I hit `C-g`. I usually wait five to ten seconds and then simply quit, using `C-g`. Not always, but usually, the page has already completed downloading and is being rendered and it therefore is not a problem.

emacs/w3 also has no bookmark functionality. This can be remedied in several ways. One simple way, mentioned below, is to use the emacs package `bmkmgr`. Another way is to use org mode with `remember` which is the method used by Dr. Raman, the author of emacspeak. See See [bmkmgr], page 40, later in this manual, for more information.

History back and next in the browser also seem to be broken but this is not generally a problem for me as I never look back.

## emacs/w3 native functions

Many functions for efficient navigation of the internet are native to emacs/w3. Some of the most useful functions are listed below

Unlike emacs-w3m, the information presented by `describe-mode` is very complete, but a little terse. All the function names are listed and asking for help on particular functions works well. That being said, use this list to augment, not replace, the built-in help that is available in emacs.

`C-f` will open a new buffer containing the cell point is in. In most instances, this works very well. Imagine you are looking at a page that is divided into 4 distinct areas: a cell at the top of the page that contains a banner and some navigation, and a "body" area that is divided into three sections consisting of more navigation, an article, and advertisement. If point is in the main article cell, using `C-f` will open another buffer that contains only the text of that cell, the article you are interested in.

One caveat is that this does not always work as advertised. Sometimes the leftmost character of each line is missing. At least it makes for interesting reading. Usually when I have this problem I simply exit that buffer and linearize the tables in the original page.

The `m` key executes a very useful command. It will complete a link on the page. Imagine that you are reading through a document and you hear a link that you need to visit.

You could tab through all the links until you hear the one you want or you could hit the *m* key and enter the link text at the prompt. Completion is available and it is not case-sensitive. Efficient, no?

The period in a cell will speak the contents of that particular cell. This command is, in my opinion, most useful when navigating tables with cells that have only one paragraph or less. I tend to not read whole articles in this manner because, inevitably, someone will interrupt me and I will lose my place.

The equals key, while in a table cell will give you the cell information. It tells you the row and column position, the size of the table, and at what nesting level the table is.

The pipe key, is used to read the table column. As this command seems to read the rectangle the column is in, this command is most useful when used in a table where there is no column spanning, i.e. all rows and columns are uniform.

Here is a list of table navigation commands:

- *C-e* + moves to the beginning of the next table row.
- *C-e* - moves to the beginning of the previous table row.
- *C-e* < moves to the beginning of the table.
- *C-e* > moves to the end of the table.
- *C-e* = moves to the top of the table column.
- *C-e* <DOWN> moves to the next cell down in the same column.
- *C-e* <UP> moves to the previous cell in the same column.
- *C-e* <LEFT> moves to the previous cell in the same row.
- *C-e* <RIGHT> moves to the next cell in the same row.

As you can see, table navigation in w3 can be easy and fun.

## emacspeak specific functions for emacs/w3

An incredible amount of work has been done by the emacspeak community to make emacs/w3 accessible to those with visual impairments. Here is an explanation of some of those functions.

The command *C-t* will toggle the visibility of table borders. This command might be useful where you want to hear all punctuation symbols on a page but the table characters get in the way.

The quote key will execute a command that allows you to skim the contents of the buffer. it will read the page, paragraph-by-paragraph, pausing between paragraphs to prompt you to move on by pressing SPACE. If you hit SPACE in the middle of a paragraph, it skips to the next paragraph.

Another skimming command is bound to the *z* key. This will allow you to zip through web pages by logical blocks such as div, paragraph, and table tags.

Using the imenu facilities is another way of skimming the document and getting to the information you desire. imenu works especially well for well-structured documents. The first thing that need be done is to copy the 'w3-imenu.el' file from the 'contrib' directory of the 'w3' directory to somewhere in your load path. I am using the cvs version of w3 and my w3 directory is under '/home/rdc/sourceforge'. The easiest

thing to do is probably to do an *M-x locate* and search for ‘w3-imenu.el’ to see where it is. After locating the file, move it into your load path. In my case I have it under ‘/home/rdc/share/emacs/site-lisp/’.

There are two ways to use the imenu facilities: automatically and manually. Since I do not use imenu on every site, I prefer to invoke it manually to save the time required to build the index.

Once things are in place, invoke imenu with the *j* key. This will ask you for an index position. Hitting TAB will give you a list of the possible index positions. Another way of navigating the document would now be to use the keys *M-n* and *M-p* to go to the next and previous index positions. Note that you have to build an index for a page before you can use these commands.

*Cap A* and *cap R* serve the same function. *Cap A* browses the Atom feed at point and *cap R* browses the rss feed at point. This is useful to sample the feed, so to speak, before going through the trouble of configuring your feed reader to fetch the feed. It might also be useful to grab the headlines from a page and present them in a more concise, readable format. If using the sort-tables xsl transform, there will be a link at the top of the page if there is an rss feed available. See See [emacspeak atom and emacspeak rss], page 40, for more information.

Google provides many useful tools for web surfers. The following commands are useful to access much google goodness.

*Cap C* extracts the current page from the google cache. With a prefix argument it will extract the link under point. This is useful for those times when a particular site is down... maybe it is in the cache... maybe it is not. It can also be used for when particular pages are removed from a site like in the case of a government conspiracy. Are we at war with Eurasia or East Asia?

The slash key will search google for pages similar to the current page.

The command *g* will do a google search restricted to the site of the document.

The *l* command googles for who links to this page.

The command *t* runs the url under point through the google transcoder. This is useful for sites that are heavy on the use of tables and the xsl transforms are not helpful. It also works on some sites that use javascript to go to the next page in the story, such as Reuters. Using a prefix argument with this command will untranscode the url under point for pages that are currently transcoded.

*Cap T* will jump to the first occurrence of the title in the document. Multiple consecutive executions of this command will jump to further occurrences. This command is probably one of the most useful timesavers while web browsing.

*M-s* jumps to the “submit” button for the form you are editing.

*M-r* plays the media stream at point with the default media player.

The *y* command will rewrite the url of the url under point. This is useful for those sites you frequent. Often, sites that have printer friendly content have a specific way in which the url is written. For example, the O’Reilly web site uses “pub” for the regular html version of a page and “lpt” for the printer friendly version. Examine these urls:

[http://www.oreillynet.com/lpt/a/wireless/2001/03/02/802.11b\\_facts.html](http://www.oreillynet.com/lpt/a/wireless/2001/03/02/802.11b_facts.html)

[http://www.oreillynet.com/lpt/a/wireless/2001/03/02/802.11b\\_facts.html](http://www.oreillynet.com/lpt/a/wireless/2001/03/02/802.11b_facts.html)

The first link goes to the regular version of the page and the second to the printer friendly version. Sometimes it is difficult to figure out the formula, so this is only useful if this is a site that you are visiting rather often, as the time to figure out the formula on a site that you rarely visit is far greater than the time you save.

The first time you run this command in a particular buffer you are prompted for a pattern to use. The pattern is in the form of

```
‘("from string" "to string")’
```

The opening paren is supplied. Remember to quote the strings or you will get an error. From this point on, until you kill the buffer in which you wrote the rule, hitting `y` on a link will use this rewrite rule to visit the page. If you mistyped the rule, providing a numeric argument will allow you to rewrite the rewrite rule. I love alliteration.

Saving the best for last, `e` is the xsl map prefix. As I mentioned in the section on `emacs-w3m`, xsl transforms are some powerful magic that takes a web page and transforms it in some way. Linearizing tables is a good example, and the one I use most often.

The keystroke `e a` prompts for an xsl transform to apply to the current page. If you know the name of the particular transform you want you can use tab completion to select it. Otherwise, you can hit `TAB` to get a buffer that contains the list of choices.

If you know that you want a particular transform done automatically you can use the command `e s` to select a transform and then `e o` to turn xslt on (the same command will turn xslt off). Then, every page opened from that point on will have the transform applied.

There is the variable `emacspeak-w3-xsl-transform` that can be set via the usual methods. This variable specifies a transform to use before displaying a web page. There is an advantage to turning on xsl transforms all the time. If you use the `‘identity.xsl’`, the `‘linearize-tables.xsl’` or the `‘sort-tables.xsl’` it can actually speed up rendering of the page. This is because the transforms provide clean and balanced html to the renderer. Additionally, using `‘sort-tables.xsl’` or `‘linearize-tables.xsl’` will provide a little more boost as rendering nested tables is particularly difficult for a web browser.

Sometimes it is just easier to read the printer friendly version of a story instead of having to linearize the tables and search for the content. Also, some sites, like the New York Times, I believe, make you navigate several pages to read the whole story, but if you select a “Print this story” link you can read the entire story on one page formatted without a lot of the cruft on the normal page. This is where the `e Cap P` command comes in. It will extract all the print streams from the current document.

Closely associated commands are `e r` and `e Cap R` which extract the media streams from the current page and from the link under point, respectively.

The command `e y` is another command that is useful for frequented sites. It does the same as the `y` command above in that it rewrites the url at point and follows it. In addition, it filters the output by a particular CSS class.

The command `e e` does more magic to the url at point. It processes the url using a specific function. For example, it can be used in retrieving radio content from the BBC. If you execute `C-e u` and type in *BBC Channel On Demand* or use tab completion to get the same, and then type in `radio4` you will be presented with a page containing a plethora of

links to other pages containing information about particular shows. On these pages there is, somewhere, a link that will play the program. If you hit enter on one of the links on the first page, you will be taken to one of these description pages. By using the `e e` command on a link you cut out this middle step and proceed directly to playing the program you are interested in.

If there is no executor defined for a current buffer, hitting TAB after `e e` will give you a list of possibilities to choose from. One nice feature of this function is that it can be used for any function. If you cannot remember the keystroke that will play the url under point in `emacspeak-m-player`, but you know what it is called, you can hit `e e` and then enter the name of the function. Nice.

The `e f` command will run the current page through an XPath filter and return the results. For more information on XPath, see <http://en.wikipedia.org/wiki/XPath>. If you wanted to see only the links on a page, when prompted enter `//a` and you will be returned every link on the page. If you wanted to see only the contents of “p” tags, you would enter `//p`. This can be useful for many things, form elements included. Giving this command a prefix argument will reverse the filter, giving you everything but the content of the specified tag.

A related command can be invoked with the `e p` keystroke. This command does the same as the filter above but works on the url under point.

## emacs/w3 tips and tricks

As I mentioned above, using `C-g` when it seems the browser is not responding will often display the page with no ill effects. Your mileage may vary. Taxes, tags and title are extra.

Also mentioned above is the use of `sort-tables` or `identity` as an xsl transform to speed up the rendering of pages. Every little bit helps.

Another useful tip is the use of the `k` key. This key will place the current url in the kill-ring for later yanking. If a page will not render correctly, using `k` will get the url and allow me to pass it to `emacs-w3m`. The counterpart to this command, `cap K` puts the url under point in the kill-ring.

### 9.5.4 Add-ons

Some of these are `emacspeak` specific, some are not. You can usually tell by the name.

## emacspeak url template

I love this package. Since changing my primary browser to `emacs/w3` I have really been giving the `url-template` package a workout. The `url-template` package contains templates that prompt you for information to supply to various sites to retrieve information without all the fuss of having to go to the site and navigate it. One really nice thing about `url-templates` is the fact that they need not be web pages. Media streams can also be made into `url-templates`.

The way to get to the templates is with the command `C-e u`. A TAB at the prompt will give you a list of the available templates. You should go now and have a look at the info manual section on See Section 11.76 [URL Templates], page 126, and read it. I'll wait here.

By the way, the ones I find most useful are the “Google Hits”, “emacswiki search” “NPR On Demand” and “Weather forecast from Weather Underground”.

## emacspeak atom and emacspeak rss

These are fairly simple rss and atom browsers for the emacspeak desktop. Using the Customize interface you add feeds in the form of titles and urls. Then you call the readers with *C-e C-u* for rss feeds and use *M-x emacspeak-atom-browse* for atom feeds. There is also emacspeak support for newsticker, an rss / atom reader that is a part of emacs 22, but I have never used it.

I personally use See Info file ‘gnus’, node ‘Top’, for rss feeds but setting gnus up for only that purpose is like hunting rabbits with a bazooka.

## emacspeak websearch

`emacspeak-websearch` provides more search options than you can shake a stick at. `emacspeak-websearch` provides search for dictionaries, news sites, software sites, google tools, weather, currency converter and much more. It can be accessed with the keystroke *C-e ?*. At the prompt, you can enter another question mark to get a list of the available search options. You will then be prompted for the necessary information. One of the nice things about this package is that, when using `w3` it attempts to jump to and read the most relevant information on the result page.

See Section 11.64 [`emacspeak-websearch`], page 104, for more information.

## bmk-mgr

This is a newcomer to the emacspeak world. In the interest of full disclosure, I am the one that wrote the emacspeak module that makes this package accessible. It is a bookmarks manager that works with both emacs/w3 and emacs-w3m. As of the writing of this document there are still some issues, especially when using it on emacs version 22, but those are being worked on. I think this is a good solution to the problem of emacs/w3 not having bookmarks functionality and providing one central bookmark location for those who regularly use both browsers. See <http://www.emacswiki.org/cgi-bin/wiki/EmacsBmkMgr>, for more information.

### 9.5.5 Conclusion

emacspeak makes the internet not only accessible to those with visual impairments, but it makes browsing and information retrieval quick and efficient. If a user will spend a little time up front to learn the tools available to access the web, the increase in efficiency and ability will more than make up for the time spent. The nice thing about these tools is that you can integrate them in your day-to-day as you have the time. While it is not necessary to use everything mentioned in the above document, if you add some of these tools to your repertoire you will not be sorry.

In the end, no one makes you use a hammer to drive nails but it sure beats using a banana.

## 9.6 Electronic Messaging Applications

Working with messaging applications involves both authoring and browsing content. Emacspeak provides a rich set of speech-enabled messaging tools. Further, all of the tools described in the previous sections integrate smoothly with the messaging applications described here; this means that you do not need to re-learn a new set of work habits when dealing with content in your messaging application.

### Related Packages

## 9.8 Editing Program Source Code

Files containing program source code form a very specific class of **structured** documents. Unlike documents meant for human consumption that are often only loosely structured, program source (as a concession to the computer's intolerance of lack of structure) are per force well-structured and adhere to a fairly stringent syntax.

The Emacs environment provides editing modes that are specific to creating and maintaining software written in most popular programming languages. Many of these editing modes are speech-enabled by Emacspeak. Speech-enabling these modes includes providing a rich set of navigational commands that allow you to move through the source efficiently. In addition, Emacspeak's core **voice-lock** facilities are used to produce audio formatted output –this helps you spot errors quickly.

### Related Packages

## 9.10 Software Development Environment

In addition to providing specialized editing modes for creating and maintaining program source, Emacs provides a rich set of software development tools that can be combined to create powerful Integrated Development Environments (IDE). These IDEs are speech-enabled by Emacspeak to provide a versatile and powerful environment for eyes-free software development.

### Related Packages

## 9.12 Desktop Management

Emacs provides an integrated environment for performing all of ones day-to-day computing tasks ranging from electronic messaging to software development. The environment derives its power from the fact that this integration allows for content to be handled across different tasks in a seamless manner. In order to work effectively with large Emacs sessions with many documents and applications open at the same time, the Emacspeak desktop provides a powerful collection of desktop management tools designed to help the user easily locate objects that pertain to a given task.

## Related Packages

### 9.14 Personal Information Management

This section describes speech-enabled tools designed to aid in personal information management such as maintaining a daily calendar.

## Related Packages

### 9.16 Desktop Applications

#### 9.16.1 Spread Sheets

Spreadsheet applications present a two dimensional view of structured data where the field values are (possibly) mutually dependent. On the Emacspeak desktop, a speech-enabled spreadsheet application can be used to manipulate such data-driven documents% ranging from simple cheque books and expense reports to complex investment portfolios. Where the traditional visual interface to spreadsheets is typically independent of the semantics of the data stored in the spreadsheet, the speech-enabled interface is derived from the *meaning* of the various fields making up the data. When presenting such information on a visual display, implicit visual layout can be used to cue the user to the meaning of different data fields. On the other hand, in the case of an actively scrolling auditory display, the spoken output needs to explicitly convey both the value and interpretation of the different data items. In addition, the interface needs to enable an active dialogue between user and application where the user is able to query the system about the possible meaning of a particular item of data. Finally, the aural interface needs to enable *multiple* views of the display. In the visual interface, such *multiple* views are automatically enabled by the two dimensional layout combined with the eye's ability to move rapidly around the layout structure. Thus, while viewing any particular row of a portfolio, one can immediately see the current total value as well as the net gain or loss. The Emacs spread-sheet package `dismal` can be retrieved from <ftp://cs.nyu.edu/pub/local/fox/dismal>.

#### 9.16.2 Forms Mode

*Forms* mode an Emacs mode designed to edit structured data records like the line shown from file `/etc/passwd` presents a user-friendly visual interface that displays the field name along with the field value. The user can edit the field value and save the file, at which point the data is written out using the underlying `:` delimited representation. Mode *forms* provides a flexible interface to associating meaning to the fields of such structured data files. For details on it use, see the forms-mode section of the online Emacs info documentation.

#### 9.16.3 OCR –Reading Print Documents

Module `emacspeak-ocr` implements an OCR front-end for the Emacspeak desktop.

Page image is acquired using tools from package `SANE` (Scanner Access Now Easy). The acquired image is run through the OCR engine if one is available, and the results

placed in a buffer that is suitable for browsing the results. This buffer is placed in mode `emacspeak-ocr-mode` a specialized mode for reading and scanning documents.

### 9.16.3.1 Emacspeak OCR Mode

Emacspeak OCR mode is a special major mode for document scanning and OCR.

Pre-requisites:

- A working scanner back-end like SANE on Linux.
- An OCR engine.

Make sure your scanner back-end works, and that you have the utilities to scan a document and acquire an image as a tiff file. Then set variable `emacspeak-ocr-scan-image-program` to point at this utility. By default, this is set to ‘scanimage’ which is the image scanning utility provided by SANE.

By default, this front-end attempts to compress the acquired tiff image; make sure you have a utility like `tiffcp`. Variable `emacspeak-ocr-compress-image` is set to ‘tiffcp’ by default; if you use something else, you should customize this variable.

Next, make sure you have an OCR engine installed and working. By default this front-end assumes that OCR is available as `/usr/bin/ocr`.

Once you have ensured that acquiring an image and applying OCR to it work independently of Emacs, you can use this Emacspeak front-end to enable easy OCR access from within Emacspeak.

The Emacspeak OCR front-end is launched by command `emacspeak-ocr` bound to `C-e C-o`.

This command switches to a special buffer that has OCR commands bounds to single keystrokes— see the key-binding list at the end of this description. Use Emacs online help facility to look up help on these commands.

Mode `emacspeak-ocr-mode` provides the necessary functionality to scan, OCR, read and save documents. By default, scanned images and the resulting text are saved under directory `~/ocr`; see variable `emacspeak-ocr-working-directory`. Invoking command `emacspeak-ocr-open-working-directory` bound to `d` will open this directory.

By default, the document being scanned is named ‘untitled’. You can name the document by using command `emacspeak-ocr-name-document` bound to `n`. The document name is used in constructing the name of the image and text files.

Here is a list of all emacspeak OCR commands along with their key-bindings and a brief description:

|                    |                                                                                                                                                                                                      |
|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>digit</code> | <code>emacspeak-ocr-page</code> Jumps to specified page in the OCR output.                                                                                                                           |
| <code>c</code>     | <code>emacspeak-ocr-set-compress-image-options</code><br>Interactively update image compression options. Prompts with current setting in the minibuffer. Setting persists for current Emacs session. |
| <code>i</code>     | <code>emacspeak-ocr-set-scan-image-options</code> Interactively update scan image options. Prompts with current setting in the minibuffer. Setting persists for current Emacs session.               |
| <code>spc</code>   | <code>emacspeak-ocr-read-current-page</code> Speaks current page.                                                                                                                                    |

|            |                                                                                           |
|------------|-------------------------------------------------------------------------------------------|
| <i>s</i>   | <i>emacspeak-ocr-save-current-page</i> Saves current page as a text file.                 |
| <i>p</i>   | <i>emacspeak-ocr-page</i> Prompts for a page number and moves to the specified page.      |
| <i>]</i>   | <i>emacspeak-ocr-forward-page</i> Move forward to the next page.                          |
| <i>[</i>   | <i>emacspeak-ocr-backward-page</i> Move back to the previous page.                        |
| <i>d</i>   | <i>emacspeak-ocr-open-working-directory</i> Open directory containing the results of OCR. |
| <i>n</i>   | <i>emacspeak-ocr-name-document</i> Name current document.                                 |
| <i>o</i>   | <i>emacspeak-ocr-recognize-image</i> Launch OCR engine on a scanned image.                |
| <i>i</i>   | <i>emacspeak-ocr-scan-image</i> Acquire an image using <code>scanimage</code> .           |
| <i>RET</i> | <i>emacspeak-ocr-scan-and-recognize</i> Scan and recognize a page.                        |
| <i>w</i>   | <i>emacspeak-ocr-write-document</i> Write all pages of current document to a text file.   |
| <i>q</i>   | <i>bury-buffer</i> Bury the OCR buffer.                                                   |
| <i>c</i>   | <i>emacspeak-ocr-customize</i> Customize Emacspeak OCR settings.                          |
| <i>?</i>   | <i>describe-mode</i> Describe OCR mode.                                                   |

## Related Packages

## 10 Running Terminal Based Applications.

You can use the terminal emulator mode to run arbitrary terminal-based programs from within Emacs. You open a terminal emulator buffer using `M-x term`, with an extra carriage return to accept the default shell (such as `bash`). (Incidentally, don't confuse this command with `M-x terminal-emulator`, which starts an older terminal emulator mode not supported by Emacspeak.)

Three kinds of commands are used within the terminal emulator. Normal term commands use a prefix of `C-c`. The emacspeak commands for `eterm` mode use a prefix of `C-t`. Anything else is a normal shell command.

There are two sub-modes of term mode: char sub-mode and line sub-mode. In char sub-mode, emacspeak will only speak the final chunk of output –typically the last line displayed. Each character typed (except ‘term-escape-char’) is sent immediately. Use char sub-mode for screen oriented programs like `vi` or `pine`.

In line sub-mode, program output is spoken if user option `eterm-autospeak` is turned on. When you type a return at the end of the buffer, that line is sent as input, while return not at end copies the rest of the line to the end and sends it. When using terminal line mode with option `eterm-autospeak` turned on, speech feedback is similar to that obtained in regular `shell-mode` buffers.

The default is char sub-mode. You can switch to line sub-mode with `C-c C-j` (recall that control J is a linefeed), and back to char sub-mode with `C-c C-k` (think of character spelled with a K).

Note: Use char-mode with the terminal emulator for running screen-oriented programs like `Lynx` or `Pine`. For regular shell interaction just use `M-x shell` instead of using the terminal emulator.

### 10.1 Char Sub-mode of Term Mode

In char sub-mode of term, each character you type is sent directly to the inferior process without intervention from emacs, except for the escape character (usually `C-c`).

Here are some of the useful commands for the char sub-mode. Note that the usual commands for killing a buffer or switching buffers do not work in this mode, so new key bindings are supplied. The first five commands are different ways of leaving this mode.

‘`C-c C-j`’

‘`M-x term-line-mode`’

Switch to line sub-mode of term mode.

‘`C-c o`’

‘`M-x other-window`’

Select the next window on this frame. All windows on current frame are arranged in a cyclic order. This command selects the next window in that order. If there are no other windows, this command does nothing.

‘`C-c C-f`’

‘`M-x find-file`’

Switch to a buffer visiting a file, creating one if none already exists.

`'C-c 0'`

`'M-x delete-window'`

Remove current window from the display.

`'C-c k'`

`'M-x kill-buffer'`

Kill the current buffer.

`'C-c C-x C-c'`

`'M-x save-buffers-kill-emacs'`

Offer to save each buffer, then kill this Emacs process.

`'C-c C-d'`

`'M-x list-directory'`

Display a list of files in or matching DIRNAME, a la 'ls'. DIRNAME is globbed by the shell if necessary. Prefix arg (C-u) means supply -l switch to 'ls'. The list appears in a second window.

`'C-c 1'`

`'M-x delete-other-windows'`

Delete all other windows in the frame, making the current window fill its frame.

`'C-c C-c'`

`'M-x term-send-raw'`

Send the last character typed through the terminal-emulator without any interpretation.

`'C-c ('`

`'M-x start-kbd-macro'`

Record subsequent keyboard input, defining a keyboard macro. The commands are recorded even as they are executed. Use C-c ) to finish recording and make the macro available. Use M-x name-last-kbd-macro to give it a permanent name. Prefix arg (C-u) means append to last macro defined; This begins by re-executing that macro as if you had typed it again.

`'C-c )'`

`'M-x end-kbd-macro'`

Finish defining a keyboard macro. The definition was started by C-c (. The macro is now available for use via C-c e, or it can be given a name with M-x name-last-kbd-macro and then invoked under that name.

`'C-c e'`

`'M-x call-last-kbd-macro'`

Call the last keyboard macro that you defined with C-c (. A prefix argument serves as a repeat count. Zero means repeat until error.

You can get a list of all the key sequences with a C-c prefix by typing C-c C-h while in this sub-mode. Some of those commands are only available in the char sub-mode, while others are generally available.

## 10.2 Line Sub-mode of Term Mode

In line sub-mode of term mode, emacs editing commands work normally, until you type RET which sends the current line to the inferior process.

Here are some of the useful commands for the line sub-mode of the term mode. In addition, the usual commands for handling a buffer work in this mode (C-x o to switch windows, C-x k to kill a buffer, C-x f to find a file, and so forth).

`'C-c C-k'`

`'M-x term-char-mode'`

Switch to char sub-mode of term mode.

`'C-c C-z'`

`'M-x term-stop-subjob'`

Stop the current subjob. Resume the subjob in the foreground with the ordinary command fg, or run it in the background with bg. WARNING: if there is no current subjob, you can end up suspending the top-level process running in the buffer. If you accidentally do this, use M-x term-continue-subjob to resume the process. (This is not a problem with with most shells, including bash, since they ignore this signal.)

`'C-c C-\'`

`'M-x term-quit-subjob'`

Send quit signal to the current subjob.

`'C-c C-c'`

`'M-x term-interrupt-subjob'`

Interrupt the current subjob.

`'C-c C-w'`

`'M-x backward-kill-word'`

Kill characters backward until encountering the end of a word.

`'C-c C-u'`

`'M-x term-kill-input'`

Kill all text from last stuff output by interpreter to point.

`'C-c C-a'`

`'M-x term-bol'`

Goes to the beginning of line, then skips past the prompt, if any. If a prefix argument is given (C-u), then no prompt skip – go straight to column 0.

`'C-c C-d'`

`'M-x term-send-eof'`

Send an end of file character (EOF) to the current buffer's process.

You can get a list of all the key sequences with a C-c prefix by typing C-c C-h while in this sub-mode. Some of those commands are only available in the line sub-mode, while others are generally available.

### 10.3 Eterm Mode Commands

The eterm mode maintains a pointer, which is not necessarily the same as the terminal's cursor. It is intended to be used in eterm's char submode. In char submode, C-t , (that's control-t followed by comma) will tell you where the eterm pointer is. C-t C-i will tell you where the terminal's cursor is. The top left corner of the window is "row 0 column 0".

The eterm pointer can be moved with C-t < (to the top of the screen), C-t > (to the bottom of the screen), C-t n (to the next line), C-t p (to the previous line), and C-t . (to the cursor). Each of these also speaks the line the pointer moves to. You can also search forward with C-t s.

These commands speak without moving the pointer: C-t l (current line), C-t w (current word), C-t c (current character), and C-t [space] (from eterm pointer to cursor).

You may enter review mode with C-t q. In review mode, you can search the buffer and speak its contents, without disturbing the terminal. Commands for moving the pointer are similar to normal editing commands, but without a control key: n and p for next and previous line, f and b for forward and back by characters, < and > for the beginning or end of the buffer. c, w, and l speak the current character, word, and line. s searches forward (not incrementally). A comma speaks the pointer location. A period moves the pointer to the terminal cursor. Return to normal term mode by typing q.

## 11 Emacspeak Commands

This chapter is generated automatically from the source-level documentation. Any errors or corrections should be made to the source-level documentation. This chapter documents a total of 763 commands.

### 11.1 cd-tool

Commentary: Provide an emacs front-end to cdtool. cdtool can be obtained as an rpm check using rpmfind or from its home site at sunsite.unc.edu /pub/Linux/apps/sound/cdrom/cli This module also provides the ability to play or save clips from a CD if you have cdda2wav installed. cdda2wav is a cd to wav convertor.

Automatically generated documentation for commands defined in module cd-tool.

**cd-tool** (*)* [Interactive Command]

*control e DEL*

Front-end to CDTool. Bind this function to a convenient key- Emacspeak users automatically have this bound to <DEL> in the emacspeak keymap.

Key Action — ——

+ Next Track - Previous Track SPC Pause or Resume e Eject = Shuffle i CD Info p Play s Stop t track c clip cap C Save clip to disk

### 11.2 dtk-speak

Commentary: Defines the TTS interface.

Automatically generated documentation for commands defined in module dtk-speak.

**dtk-add-cleanup-pattern** (**&optional DELETE**) [Interactive Command]

*control e d a*

Add this pattern to the list of repeating patterns that are cleaned up. Optional interactive prefix arg deletes this pattern if previously added. Cleaning up repeated patterns results in emacspeak speaking the pattern followed by a repeat count instead of speaking all the characters making up the pattern. Thus, by adding the repeating pattern ‘.’ (this is already added by default) emacspeak will say “aw fifteen dot” when speaking the string “.....” instead of “period period period period ”

**dtk-pause** (**&optional PREFIX**) [Interactive Command]

*control e p*

Pause ongoing speech. The speech can be resumed with command ‘dtk-resume’ normally bound to C-e SPC. Pausing speech is useful when one needs to perform a few actions before continuing to read a large document. Emacspeak gives you speech feedback as usual once speech has been paused. ‘dtk-resume’ continues the interrupted speech irrespective of the buffer in which it is executed. Optional PREFIX arg flushes any previously paused speech.

- dtk-reset-state** () [Interactive Command]  
*control e d cap R*  
 Restore sanity to the Dectalk. Typically used after the Dectalk has been power cycled.
- dtk-resume** () [Interactive Command]  
*control e SPACE*  
 Resume paused speech. This command resumes speech that has been suspended by executing command ‘dtk-pause’ bound to C-e p. If speech has not been paused, and option ‘dtk-resume-should-toggle’ is set, then this command will pause ongoing speech.
- dtk-select-server** (*PROGRAM*) [Interactive Command]  
*control e d d*  
 Select a speech server interactively. Argument *PROGRAM* specifies the speech server program. When called interactively, The selected server is started immediately.
- dtk-set-character-scale** (*FACTOR &optional PREFIX*) [Interactive Command]  
*control e d f*  
 Set scale *FACTOR* for speech rate. Speech rate is scaled by this factor when speaking characters. Interactive *PREFIX* arg means set the global default value, and then set the current local value to the result.
- dtk-set-chunk-separator-syntax** (*S*) [Interactive Command]  
*control e d RETURN*  
 Interactively set how text is split in chunks. See the Emacs documentation on syntax tables for details on how characters are classified into various syntactic classes. Argument *S* specifies the syntax class.
- dtk-set-language** (*LANG*) [Interactive Command]  
 Set language according to the argument lang.
- dtk-set-next-language** () [Interactive Command]  
 Switch to the next available language
- dtk-set-predefined-speech-rate** (**&optional** *PREFIX*) [Interactive Command]  
*control e d 0 control e d 1 control e d 2 control e d 3 control e d 4 control e d 5 control e d 6 control e d 7 control e d 8 control e d 9*  
 Set speech rate to one of nine predefined levels. Interactive *PREFIX* arg says to set the rate globally. Formula used is: rate = dtk-speech-rate-base + dtk-speech-rate-step \* level.
- dtk-set-preferred-language** (*ALIAS LANG*) [Interactive Command]  
 Set the alias of the preferred language: For example if alias="en" lang="en-GB", then the following call: dtk-set-language("en") will set "en-GB".

- dtk-set-previous-language** () [Interactive Command]  
Switch to the previous available language
- dtk-set-punctuations** (*MODE* &optional *PREFIX*) [Interactive Command]  
*control e d p*  
Set punctuation mode to *MODE*. Possible values are ‘some’, ‘all’, or ‘none’. Interactive *PREFIX* arg means set the global default value, and then set the current local value to the result.
- dtk-set-punctuations-to-all** (&optional *PREFIX*) [Interactive Command]  
Set punctuation mode to all. Interactive *PREFIX* arg sets punctuation mode globally.
- dtk-set-punctuations-to-some** (&optional *PREFIX*) [Interactive Command]  
Set punctuation mode to some. Interactive *PREFIX* arg sets punctuation mode globally.
- dtk-set-rate** (*RATE* &optional *PREFIX*) [Interactive Command]  
*control e d r*  
Set speaking *RATE* for the tts. Interactive *PREFIX* arg means set the global default value, and then set the current local value to the result.
- dtk-stop** () [Interactive Command]  
*<pause> control e s*  
Stop speech now.
- dtk-toggle-allcaps-beep** (&optional *PREFIX*) [Interactive Command]  
*control e d cap C*  
Toggle allcaps-beep. when set, allcaps words are indicated by a short beep. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result. Note that allcaps-beep is a very useful thing when programming. However it is irritating to have it on when reading documents.
- dtk-toggle-capitalization** (&optional *PREFIX*) [Interactive Command]  
*control e d c*  
Toggle capitalization. when set, capitalization is indicated by a short beep. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result.
- dtk-toggle-debug** (&optional *PREFIX*) [Interactive Command]  
*control e d b*  
Toggle state of the debug *FLAG*. When debugging is on, you can switch to the buffer *\*speaker\** to examine the output from the process that talks to the speech device by using command *C-e d C-M-b*. Note: *\*speaker\** is a hidden buffer, ie it has a leading space in its name.
- dtk-toggle-punctuation-mode** (&optional *PREFIX*) [Interactive Command]  
Toggle punctuation mode between "some" and "all". Interactive *PREFIX* arg makes the new setting global.

**dtk-toggle-quiet** (**&optional** *PREFIX*) [Interactive Command]  
*control e d q*

Toggles state of dtk-quiet. Turning on this switch silences speech. Optional interactive prefix arg causes this setting to become global.

**dtk-toggle-speak-nonprinting-chars** (**&optional** *PREFIX*) [Interactive Command]  
*control e d n*

Toggle speak-nonprinting-chars. Switches behavior of how characters with the high bit set are handled. Interactive PREFIX arg means toggle the global default value, and then set the current local value to the result.

**dtk-toggle-split-caps** (**&optional** *PREFIX*) [Interactive Command]  
*control e d s*

Toggle split caps mode. Split caps mode is useful when reading Hungarian notation in program source code. Interactive PREFIX arg means toggle the global default value, and then set the current local value to the result.

**dtk-toggle-splitting-on-white-space** () [Interactive Command]  
*control e d SPACE*

Toggle splitting of speech on white space. This affects the internal state of emacspeak that decides if we split text purely by clause boundaries, or also include white space. By default, emacspeak sends a clause at a time to the speech device. This produces fluent speech for normal use. However in modes such as ‘shell-mode’ and some programming language modes, clause markers appear infrequently, and this can result in large amounts of text being sent to the speech device at once, making the system unresponsive when asked to stop talking. Splitting on white space makes emacspeak’s stop command responsive. However, when splitting on white space, the speech sounds choppy since the synthesizer is getting a word at a time.

**dtk-toggle-stop-immediately-while-typing** [Interactive Command]  
 (**&optional** *PREFIX*)  
*control e d cap I*

Toggle state of variable ‘dtk-stop-immediately-while-typing’. As the name implies, if T then speech flushes immediately as you type. Optional argument PREFIX specifies if the setting applies to all buffers.

**dtk-toggle-strip-octals** (**&optional** *PREFIX*) [Interactive Command]  
*control e d o*

Toggle stripping of octals. Interactive prefix arg means toggle the global default value, and then set the current local value to the result.

### 11.3 emacspeak-alsaplayer

Commentary: Defines a simple derived mode for interacting with alsaplayer. alsaplayer navigation commands work via single keystrokes.

Automatically generated documentation for commands defined in module emacspeak-alsaplayer.

|                                                                                                        |                       |
|--------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-alsaplayer-add-to-queue</code> ( <i>RESOURCE</i> )                                     | [Interactive Command] |
| Add specified resource to queue.                                                                       |                       |
| <code>emacspeak-alsaplayer-backward-10-seconds</code> ()                                               | [Interactive Command] |
| Skip backward by 10 seconds.                                                                           |                       |
| <code>emacspeak-alsaplayer-backward-minute</code> ( <i>MINUTES</i> )                                   | [Interactive Command] |
| Skip backwards by minutes.                                                                             |                       |
| <code>emacspeak-alsaplayer-backward-ten-minutes</code><br>( <i>MINUTES</i> )                           | [Interactive Command] |
| Skip backwards by chunks of minutes.                                                                   |                       |
| <code>emacspeak-alsaplayer-clear</code> ()                                                             | [Interactive Command] |
| Clear or resume alsaplayer                                                                             |                       |
| <code>emacspeak-alsaplayer-clip</code> ( <i>PATH START END</i> )                                       | [Interactive Command] |
| Invoke mp3splt to clip selected range.                                                                 |                       |
| <code>emacspeak-alsaplayer-forward-10-seconds</code> ()                                                | [Interactive Command] |
| Skip forward by seconds.                                                                               |                       |
| <code>emacspeak-alsaplayer-forward-minute</code> ( <i>MINUTES</i> )                                    | [Interactive Command] |
| Skip forward by minutes.                                                                               |                       |
| <code>emacspeak-alsaplayer-forward-ten-minutes</code><br>( <i>MINUTES</i> )                            | [Interactive Command] |
| Skip forward by chunks of ten minutes.                                                                 |                       |
| <code>emacspeak-alsaplayer-info</code> ()                                                              | [Interactive Command] |
| Speak current path and copy it to kill ring.                                                           |                       |
| <code>emacspeak-alsaplayer-jump</code> ( <i>TRACK</i> )                                                | [Interactive Command] |
| Jump to specified track.                                                                               |                       |
| <code>emacspeak-alsaplayer-launch</code> ()                                                            | [Interactive Command] |
| Launch Alsaplayer. user is placed in a buffer associated with the newly created Alsaplayer session.    |                       |
| <code>emacspeak-alsaplayer-mark-position</code> ()                                                     | [Interactive Command] |
| Mark currently displayed position.                                                                     |                       |
| <code>emacspeak-alsaplayer-mode</code> ()                                                              | [Interactive Command] |
| Major mode for alsaplayer interaction.                                                                 |                       |
| key binding — ——                                                                                       |                       |
| SPC <code>emacspeak-alsaplayer-pause</code> , <code>emacspeak-alsaplayer-backward-10-seconds</code>    |                       |
| . <code>emacspeak-alsaplayer-forward-10-seconds</code> / <code>emacspeak-alsaplayer-speed</code> <     |                       |
| <code>emacspeak-alsaplayer-backward-minute</code> > <code>emacspeak-alsaplayer-forward-minute</code>   |                       |
| ? <code>emacspeak-alsaplayer-status</code> A <code>emacspeak-alsaplayer-replace-queue</code> S         |                       |
| <code>emacspeak-alsaplayer-stop</code> [ <code>emacspeak-alsaplayer-backward-ten-minutes</code> ]      |                       |
| <code>emacspeak-alsaplayer-forward-ten-minutes</code> a <code>emacspeak-alsaplayer-add-to-queue</code> |                       |

c emacspeak-alsaplayer-clear g emacspeak-alsaplayer-seek i emacspeak-alsaplayer-info j emacspeak-alsaplayer-jump l emacspeak-alsaplayer-launch m emacspeak-alsaplayer-mark-position n emacspeak-alsaplayer-next p emacspeak-alsaplayer-previous q emacspeak-alsaplayer-quit r emacspeak-alsaplayer-relative s emacspeak-alsaplayer-start v emacspeak-alsaplayer-volume w emacspeak-alsaplayer-where x emacspeak-alsaplayer-clip

This mode runs the hook ‘emacspeak-alsaplayer-mode-hook’, as the final step during initialization.

|                                                                   |                       |
|-------------------------------------------------------------------|-----------------------|
| <code>emacspeak-alsaplayer-next ()</code>                         | [Interactive Command] |
| Next alsaplayer                                                   |                       |
| <code>emacspeak-alsaplayer-pause ()</code>                        | [Interactive Command] |
| Pause or resume alsaplayer                                        |                       |
| <code>emacspeak-alsaplayer-previous ()</code>                     | [Interactive Command] |
| Previous alsaplayer                                               |                       |
| <code>emacspeak-alsaplayer-quit ()</code>                         | [Interactive Command] |
| Quit or resume alsaplayer                                         |                       |
| <code>emacspeak-alsaplayer-relative (<i>OFFSET</i>)</code>        | [Interactive Command] |
| Relative seek alsaplayer                                          |                       |
| <code>emacspeak-alsaplayer-replace-queue (<i>RESOURCE</i>)</code> | [Interactive Command] |
| Replace currently playing music.                                  |                       |
| <code>emacspeak-alsaplayer-seek (<i>OFFSET</i>)</code>            | [Interactive Command] |
| Absolute seek alsaplayer                                          |                       |
| <code>emacspeak-alsaplayer-speed (<i>SETTING</i>)</code>          | [Interactive Command] |
| Set speed in alsaplayer.                                          |                       |
| <code>emacspeak-alsaplayer-start ()</code>                        | [Interactive Command] |
| Start alsaplayer                                                  |                       |
| <code>emacspeak-alsaplayer-status ()</code>                       | [Interactive Command] |
| Show alsaplayer status                                            |                       |
| <code>emacspeak-alsaplayer-stop ()</code>                         | [Interactive Command] |
| Stop alsaplayer                                                   |                       |
| <code>emacspeak-alsaplayer-volume (<i>SETTING</i>)</code>         | [Interactive Command] |
| Set volume.                                                       |                       |
| <code>emacspeak-alsaplayer-where ()</code>                        | [Interactive Command] |
| Speak current position and copy it to kill ring.                  |                       |

## 11.4 emacspeak-amphetadesk

Automatically generated documentation for commands defined in module emacspeak-amphetadesk.

**emacspeak-amphetadesk** (**&optional** *USE-OPML*) [Interactive Command]  
 Open amphetadesk. Interactive prefix-arg use-opml opens the myChannels.opml file.

**emacspeak-amphetadesk-quick-add** (*URL*) [Interactive Command]  
 Quick add URL to Amphetadesk by prompting for URL.

## 11.5 emacspeak-arc

Automatically generated documentation for commands defined in module emacspeak-arc.

**emacspeak-arc-speak-file-modification-time** () [Interactive Command]  
 Speak modification time of the file on current line

**emacspeak-arc-speak-file-name** () [Interactive Command]  
 Speak the name of the file on current line

**emacspeak-arc-speak-file-permissions** () [Interactive Command]  
 Speak permissions of file current entry

**emacspeak-arc-speak-file-size** () [Interactive Command]  
 Speak the size of the file on current line

## 11.6 emacspeak-aumix

Commentary: Provides an AUI to setting up the auditory display via AUMIX This module is presently Linux specific

Automatically generated documentation for commands defined in module emacspeak-aumix.

**emacspeak-aumix** () [Interactive Command]  
*control e* (  
 Setup output parameters of the auditory display. Launch this tool while you have auditory output on multiple channels playing so you can adjust the settings to your preference. Hit q to quit when you are done.

**emacspeak-aumix-edit** () [Interactive Command]  
 Edit aumix settings interactively. Run command M-x emacspeak-aumix-reset after saving the settings to have them take effect.

**emacspeak-aumix-reset** () [Interactive Command]  
 Reset to default audio settings.

**emacspeak-aumix-volume-decrease** (**&optional** *GAIN*) [Interactive Command]  
 Decrease overall volume.

**emacspeak-aumix-volume-increase** (*&optional GAIN*) [Interactive Command]  
 Increase overall volume.

**emacspeak-aumix-wave-decrease** (*&optional GAIN*) [Interactive Command]  
 Decrease volume of wave output.

**emacspeak-aumix-wave-increase** (*&optional GAIN*) [Interactive Command]  
 Increase volume of wave output.

## 11.7 emacspeak-bs

Commentary: speech-enable bs.el this is an alternative to list-buffers

Automatically generated documentation for commands defined in module emacspeak-bs.

**emacspeak-bs-speak-buffer-line** () [Interactive Command]  
 Speak information about this buffer

## 11.8 emacspeak-buff-menu

Automatically generated documentation for commands defined in module emacspeak-buff-menu.

**emacspeak-list-buffers-next-line** (*COUNT*) [Interactive Command]  
 Speech enabled buffer menu navigation

**emacspeak-list-buffers-previous-line** (*COUNT*) [Interactive Command]  
 Speech enabled buffer menu navigation

**emacspeak-list-buffers-speak-buffer-line** () [Interactive Command]  
 Speak information about this buffer

**emacspeak-list-buffers-speak-buffer-name** () [Interactive Command]  
 Speak the name of the buffer on this line

## 11.9 emacspeak-c

Commentary: Make some of C and C++ mode more emacspeak friendly Works with both boring c-mode and the excellent cc-mode

Automatically generated documentation for commands defined in module emacspeak-c.

**emacspeak-c-speak-semantic** () [Interactive Command]  
 Speak the C semantics of this line.

## 11.10 emacspeak-calendar

Automatically generated documentation for commands defined in module emacspeak-calendar.

**emacspeak-appt-repeat-announcement** () [Interactive Command]  
*control e cap A*

Speaks the most recently displayed appointment message if any.

**emacspeak-calendar-speak-date** () [Interactive Command]  
 Speak the date under point when called in Calendar Mode.

## 11.11 emacspeak-compile

Commentary: This module makes compiling code from inside Emacs speech friendly. It is an example of how a little amount of code can make Emacspeak even better.

Automatically generated documentation for commands defined in module emacspeak-compile.

**emacspeak-compilation-speak-error** () [Interactive Command]  
 Speech feedback about the compilation error.

## 11.12 emacspeak-custom

Automatically generated documentation for commands defined in module emacspeak-custom.

**emacspeak-custom-goto-group** () [Interactive Command]  
 Jump to custom group when in a customization buffer.

**emacspeak-custom-goto-toolbar** () [Interactive Command]  
 Jump to custom toolbar when in a customization buffer.

## 11.13 emacspeak-daisy

Commentary: Daisy Digital Talking Book Reader

Automatically generated documentation for commands defined in module emacspeak-daisy.

**emacspeak-daisy-define-outline-pattern** (*REGEXP*) [Interactive Command]  
 Define persistent outline regexp for this book.

**emacspeak-daisy-mark-position-in-content-under-point** () [Interactive Command]  
 ()  
 Mark current position in displayed content. No-op if content under point is not currently displayed.

**emacspeak-daisy-mode** () [Interactive Command]  
 A DAISY front-end for the Emacspeak desktop.  
 Pre-requisites:

0) mpg123 for playing mp3 files 1) libxml and libxslt packages 2) xml-parse.el for parsing XML in Emacs Lisp.

The Emacspeak DAISY front-end is launched by command `emacspeak-daisy-open-book` bound to C-e C-b.

This command switches to a special buffer that has DAISY commands bounds to single keystrokes— see the ke-binding list at the end of this description. Use Emacs online help facility to look up help on these commands.

`emacspeak-daisy-mode` provides the necessary functionality to navigate and listen to Daisy talking books.

Here is a list of all emacspeak DAISY commands along with their key-bindings:

key binding — ——

RET `emacspeak-daisy-play-content-under-point` SPC `emacspeak-daisy-play-audio-under-point` ? `describe-mode` P `emacspeak-daisy-play-page-range` S `emacspeak-daisy-save-bookmarks` m `emacspeak-daisy-mark-position-in-content-under-point` n `emacspeak-daisy-next-line` o `emacspeak-daisy-define-outline-pattern` p `emacspeak-daisy-previous-line` q `bury-buffer` s `emacspeak-daisy-stop-audio`

In addition to any hooks its parent mode ‘text-mode’ might have run, this mode runs the hook ‘`emacspeak-daisy-mode-hook`’, as the final step during initialization.

|                                                                                                                                                                                      |                       |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-daisy-next-line</code> ()                                                                                                                                            | [Interactive Command] |
| Move to next line.                                                                                                                                                                   |                       |
| <code>emacspeak-daisy-open-book</code> ( <i>FILENAME</i> )                                                                                                                           | [Interactive Command] |
| <i>control e control b</i>                                                                                                                                                           |                       |
| Open Digital Talking Book specified by navigation file filename.                                                                                                                     |                       |
| This is the main entry point to the Emacspeak Daisy reader. Opening a Daisy navigation file (.ncx file) results in a navigation buffer that can be used to browse and read the book. |                       |
| <code>emacspeak-daisy-play-audio-under-point</code> ()                                                                                                                               | [Interactive Command] |
| Play audio clip under point.                                                                                                                                                         |                       |
| <code>emacspeak-daisy-play-content-under-point</code> ()                                                                                                                             | [Interactive Command] |
| Play SMIL content under point.                                                                                                                                                       |                       |
| <code>emacspeak-daisy-play-page-range</code> ( <i>START END</i> )                                                                                                                    | [Interactive Command] |
| Play pages in specified page range.                                                                                                                                                  |                       |
| <code>emacspeak-daisy-previous-line</code> ()                                                                                                                                        | [Interactive Command] |
| Move to previous line.                                                                                                                                                               |                       |
| <code>emacspeak-daisy-save-bookmarks</code> ()                                                                                                                                       | [Interactive Command] |
| Save bookmarks for current book.                                                                                                                                                     |                       |
| <code>emacspeak-daisy-stop-audio</code> ()                                                                                                                                           | [Interactive Command] |
| Stop audio.                                                                                                                                                                          |                       |

## 11.14 emacspeak-dired

Commentary: This module speech enables dired. It reduces the amount of speech you hear: Typically you hear the file names as you move through the dired buffer. Voicification is used to indicate directories, marked files etc.

Automatically generated documentation for commands defined in module emacspeak-dired.

**emacspeak-dired-label-fields** () [Interactive Command]  
 Labels the fields of the listing in the dired buffer. Currently is a no-op unless unless 'dired-listing-switches' contains -al

**emacspeak-dired-show-file-type** (&optional *FILE* [Interactive Command]  
*DEREF-SYMLINKS*)

Displays type of current file by running command file. Like Emacs' built-in dired-show-file-type but allows user to customize options passed to command 'file'.

**emacspeak-dired-speak-file-access-time** () [Interactive Command]  
 Speak access time of the current file.

**emacspeak-dired-speak-file-modification-time** () [Interactive Command]  
 Speak modification time of the current file.

**emacspeak-dired-speak-file-permissions** () [Interactive Command]  
 Speak the permissions of the current file.

**emacspeak-dired-speak-file-size** () [Interactive Command]  
 Speak the size of the current file. On a directory line, run du -s on the directory to speak its size.

**emacspeak-dired-speak-header-line** () [Interactive Command]  
 Speak the header line of the dired buffer.

**emacspeak-dired-speak-symlink-target** () [Interactive Command]  
 Speaks the target of the symlink on the current line.

## 11.15 emacspeak-dismal

Automatically generated documentation for commands defined in module emacspeak-dismal.

**emacspeak-dismal-backward-col-and-summarize** [Interactive Command]  
 (*COLS*)

Move backward by arg columns (the previous column by default)and summarize it.

**emacspeak-dismal-backward-row-and-summarize** [Interactive Command]  
 (*ROWS*)

Move backward by arg rows (the previous row by default)and summarize it.

**emacspeak-dismal-col-summarize** () [Interactive Command]  
 Summarizes a col using the specification in list emacspeak-dismal-col-summarizer-list

- `emacspeak-dismal-display-cell-expression ()` [Interactive Command]  
 Display the expression in the message area
- `emacspeak-dismal-display-cell-value ()` [Interactive Command]  
 Display the cell value in the message area
- `emacspeak-dismal-display-cell-with-col-header ()` [Interactive Command]  
 Display current cell along with its column header. The ‘column header’ is the entry in row 0.
- `emacspeak-dismal-display-cell-with-row-header ()` [Interactive Command]  
 Displays current cell along with its row header. The ‘row header’ is the entry in column 0.
- `emacspeak-dismal-forward-col-and-summarize (COLS)` [Interactive Command]  
 Move forward by arg columns (the next column by default)and summarize it.
- `emacspeak-dismal-forward-row-and-summarize (ROWS)` [Interactive Command]  
 Move forward by arg rows (the next row by default)and summarize it.
- `emacspeak-dismal-row-summarize ()` [Interactive Command]  
 Summarizes a row using the specification in list `emacspeak-dismal-row-summarizer-list`
- `emacspeak-dismal-set-col-summarizer-list ()` [Interactive Command]  
 Specify or reset col summarizer list.
- `emacspeak-dismal-set-row-summarizer-list ()` [Interactive Command]  
 Specify or reset row summarizer list.
- `emacspeak-dismal-set-sheet-summarizer-list ()` [Interactive Command]  
 Specify or reset sheet summarizer list.
- `emacspeak-dismal-sheet-summarize ()` [Interactive Command]  
 Summarizes a sheet using the specification in list `emacspeak-dismal-sheet-summarizer-list`

## 11.16 emacspeak-ecb

Automatically generated documentation for commands defined in module `emacspeak-ecb`.

- `emacspeak-ecb-speak-window-directories ()` [Interactive Command]  
 Speak contents of directories window.
- `emacspeak-ecb-speak-window-history ()` [Interactive Command]  
 Speak contents of history window.
- `emacspeak-ecb-speak-window-methods ()` [Interactive Command]  
 Speak contents of methods window.

`emacspeak-ecb-speak-window-sources ()` [Interactive Command]  
 Speak contents of sources window.

`emacspeak-ecb-tree-backspace ()` [Interactive Command]  
 Back up during incremental search in tree buffers.

`emacspeak-ecb-tree-clear ()` [Interactive Command]  
 Clear search pattern during incremental search in tree buffers.

`emacspeak-ecb-tree-expand-common-prefix ()` [Interactive Command]  
 Expand to longest common prefix in tree buffer.

`emacspeak-ecb-tree-shift-return ()` [Interactive Command]  
 Do shift return in ECB tree browser.

### 11.17 emacspeak-ediff

Automatically generated documentation for commands defined in module emacspeak-ediff.

`emacspeak-ediff-speak-current-difference ()` [Interactive Command]  
 Speak the current difference

### 11.18 emacspeak-enriched

Automatically generated documentation for commands defined in module emacspeak-enriched.

`emacspeak-enriched-voiceify-faces (START END)` [Interactive Command]  
 Map base fonts to voices. Useful in voicifying rich text.

### 11.19 emacspeak-entertain

Commentary: Auditory interface to misc games

Automatically generated documentation for commands defined in module emacspeak-entertain.

`emacspeak-hangman-speak-guess ()` [Interactive Command]  
 Speak current guessed string.

`emacspeak-hangman-speak-statistics ()` [Interactive Command]  
 Speak statistics.

### 11.20 emacspeak-eperiodic

Automatically generated documentation for commands defined in module emacspeak-eperiodic.

`emacspeak-eperiodic-goto-property-section ()` [Interactive Command]  
 Mark position and jump to properties section.

|                                                           |                       |
|-----------------------------------------------------------|-----------------------|
| <code>emacspeak-eperiodic-next-line ()</code>             | [Interactive Command] |
| Move to next row and speak element.                       |                       |
| <code>emacspeak-eperiodic-play-description ()</code>      | [Interactive Command] |
| Play audio description from WebElements.                  |                       |
| <code>emacspeak-eperiodic-previous-line ()</code>         | [Interactive Command] |
| Move to next row and speak element.                       |                       |
| <code>emacspeak-eperiodic-speak-current-element ()</code> | [Interactive Command] |
| Speak element at point.                                   |                       |

## 11.21 emacspeak-erc

Commentary: `erc.el` is a modern Emacs client for IRC including color and font locking support. `erc.el` - an Emacs IRC client (by Alexander L. Belikoff)  
<http://www.cs.cmu.edu/~berez/irc/erc.el>

Automatically generated documentation for commands defined in module `emacspeak-erc`.

|                                                                                                                                                                             |                       |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-erc-add-name-to-monitor (<i>NAME</i><br/>&amp;optional <i>QUITEN-PRONUNCIATION</i>)</code>                                                                  | [Interactive Command] |
| Add people to monitor in this room. Optional interactive prefix arg defines a pronunciation that silences speaking of this perso's name.                                    |                       |
| <code>emacspeak-erc-delete-name-from-monitor (<i>NAME</i>)</code>                                                                                                           | [Interactive Command] |
| Remove name to monitor in this room.                                                                                                                                        |                       |
| <code>emacspeak-erc-setup-cricket-rules ()</code>                                                                                                                           | [Interactive Command] |
| Set up <code>#cricket</code> channels.                                                                                                                                      |                       |
| <code>emacspeak-erc-toggle-my-monitor (&amp;optional<br/><i>PREFIX</i>)</code>                                                                                              | [Interactive Command] |
| Toggle state of ERC monitor of my messages. Interactive <code>PREFIX</code> arg means toggle the global default value, and then set the current local value to the result.  |                       |
| <code>emacspeak-erc-toggle-room-monitor (&amp;optional<br/><i>PREFIX</i>)</code>                                                                                            | [Interactive Command] |
| Toggle state of ERC room monitor. Interactive <code>PREFIX</code> arg means toggle the global default value, and then set the current local value to the result.            |                       |
| <code>emacspeak-erc-toggle-speak-all-participants<br/>(&amp;optional <i>PREFIX</i>)</code>                                                                                  | [Interactive Command] |
| Toggle state of ERC speak all participants.. Interactive <code>PREFIX</code> arg means toggle the global default value, and then set the current local value to the result. |                       |

## 11.22 emacspeak-eterm

Commentary: This module makes eterm talk. Eterm is the new terminal emulator for Emacs. Use of emacspeak with eterm really needs an info page. At present, the only documentation is the source level documentation. This module uses Control-t as an additional prefix key to allow the user To move around the terminal and have different parts spoken.

Automatically generated documentation for commands defined in module emacspeak-eterm.

**emacspeak-eterm-copy-region-to-register** [Interactive Command]  
(*REGISTER*)

Copy text from terminal to an Emacs REGISTER. This copies region delimited by the emacspeak eterm marker set by command M-x emacspeak-eterm-set-marker and the emacspeak eterm pointer to a register.

**emacspeak-eterm-define-window** (*ID*) [Interactive Command]

Prompt for a window ID. The window is then define to be the rectangle delimited by point and eterm mark. This is to be used when emacspeak is set to review mode inside an eterm.

**emacspeak-eterm-describe-window** (*ID*) [Interactive Command]

Describe an eterm window. Description indicates eterm window coordinates and whether it is stretchable

**emacspeak-eterm-goto-line** (*LINE*) [Interactive Command]

Move emacspeak eterm pointer to a specified LINE.

**emacspeak-eterm-kill-ring-save-region** () [Interactive Command]

Copy text from terminal to kill ring. This copies region delimited by the emacspeak eterm marker set by command M-x emacspeak-eterm-set-marker and the emacspeak eterm pointer.

**emacspeak-eterm-maybe-send-raw** () [Interactive Command]

Send a raw character through if in the terminal buffer. Execute end of line if in a non eterm buffer if executed via C-e C-e

**emacspeak-eterm-paste-register** (*REGISTER*) [Interactive Command]

Paste contents of REGISTER at current location. If the specified register contains text, then that text is sent to the terminal as if it were typed by the user.

**emacspeak-eterm-pointer-backward-word** (*COUNT*) [Interactive Command]

Move the pointer backward by words. Interactive numeric prefix arg specifies number of words to move. Argument COUNT specifies number of words by which to move.

**emacspeak-eterm-pointer-down** (*COUNT*) [Interactive Command]

Move the pointer down a line. Argument COUNT specifies number of lines by which to move.

- `emacspeak-eterm-pointer-forward-word` (*COUNT*) [Interactive Command]  
Move the pointer forward by words. Interactive numeric prefix arg specifies number of words to move. Argument *COUNT* specifies number of words by which to move.
- `emacspeak-eterm-pointer-left` (*COUNT*) [Interactive Command]  
Move the pointer left. Argument *COUNT* specifies number of columns by which to move.
- `emacspeak-eterm-pointer-right` (*COUNT*) [Interactive Command]  
Move the pointer right. Argument *COUNT* specifies number of columns by which to move.
- `emacspeak-eterm-pointer-to-bottom` () [Interactive Command]  
Move the pointer to the bottom of the screen.
- `emacspeak-eterm-pointer-to-cursor` () [Interactive Command]  
Move the pointer to the cursor.
- `emacspeak-eterm-pointer-to-left-edge` () [Interactive Command]  
Move the pointer to the right edge.
- `emacspeak-eterm-pointer-to-next-color-change` [Interactive Command]  
(**&optional** *COUNT*)  
Move the eterm pointer to the next color change. This allows you to move between highlighted regions of the screen. Optional argument *COUNT* specifies how many changes to skip.
- `emacspeak-eterm-pointer-to-previous-color-change` [Interactive Command]  
(**&optional** *COUNT*)  
Move the eterm pointer to the next color change. This allows you to move between highlighted regions of the screen. Optional argument *COUNT* specifies how many changes to skip.
- `emacspeak-eterm-pointer-to-right-edge` () [Interactive Command]  
Move the pointer to the right edge.
- `emacspeak-eterm-pointer-to-top` () [Interactive Command]  
Move the pointer to the top of the screen.
- `emacspeak-eterm-pointer-up` (*COUNT*) [Interactive Command]  
Move the pointer up a line. Argument *COUNT* specifies number of lines by which to move.
- `emacspeak-eterm-remote-term` (*HOST*) [Interactive Command]  
*control e CONTROL meta r*  
Start a terminal-emulator in a new buffer.
- `emacspeak-eterm-search-backward` () [Interactive Command]  
Search backward on the terminal.
- `emacspeak-eterm-search-forward` () [Interactive Command]  
Search forward on the terminal.

- `emacspeak-eterm-set-filter-window` (*FLAG*) [Interactive Command]  
 Prompt for the id of a predefined window, and set the ‘filter’ window to it. Non-nil interactive prefix arg ‘unsets’ the filter window; this is equivalent to having the entire terminal as the filter window (this is what `eterm` starts up with). Setting the filter window results in emacspeak only monitoring screen activity within the filter window.
- `emacspeak-eterm-set-focus-window` (*FLAG*) [Interactive Command]  
 Prompt for the id of a predefined window, and set the ‘focus’ window to it. Non-nil interactive prefix arg ‘unsets’ the focus window; this is equivalent to having the entire terminal as the focus window (this is what `eterm` starts up with). Setting the focus window results in emacspeak monitoring screen and speaking that window upon seeing screen activity.
- `emacspeak-eterm-set-marker` () [Interactive Command]  
 Set Emacspeak `eterm` marker. This sets the emacspeak `eterm` marker to the position pointed to by the emacspeak `eterm` pointer.
- `emacspeak-eterm-speak-cursor` () [Interactive Command]  
 Speak cursor position.
- `emacspeak-eterm-speak-pointer` () [Interactive Command]  
 Speak current pointer position.
- `emacspeak-eterm-speak-pointer-char` (&optional *PREFIX*) [Interactive Command]  
 Speak char under `eterm` pointer. Pronounces character phonetically unless called with a *PREFIX* arg.
- `emacspeak-eterm-speak-pointer-line` () [Interactive Command]  
 Speak the line the pointer is on.
- `emacspeak-eterm-speak-pointer-word` () [Interactive Command]  
 Speak the word the pointer is on.
- `emacspeak-eterm-speak-predefined-window` () [Interactive Command]  
 Speak a predefined `eterm` window between 1 and 10.
- `emacspeak-eterm-speak-screen` (&optional *FLAG*) [Interactive Command]  
 Speak the screen. Default is to speak from the emacspeak pointer to point. Optional prefix arg *FLAG* causes region above the Emacspeak pointer to be spoken.
- `emacspeak-eterm-speak-window` (*ID*) [Interactive Command]  
 Speak an `eterm` window. Argument *ID* specifies the window.
- `emacspeak-eterm-toggle-filter-window` () [Interactive Command]  
 Toggle active state of filter window.
- `emacspeak-eterm-toggle-focus-window` () [Interactive Command]  
 Toggle active state of focus window.

**emacspeak-eterm-toggle-pointer-mode** (**&optional** *PREFIX*) [Interactive Command]

Toggle emacspeak eterm pointer mode. With optional interactive prefix arg, turn it on. When emacspeak eterm is in pointer mode, the eterm read pointer stays where it is rather than automatically moving to the terminal cursor when there is terminal activity.

**emacspeak-eterm-toggle-review** () [Interactive Command]

Toggle state of eterm review. In review mode, you can move around the terminal and listen to the contnets without sending input to the terminal itself.

**emacspeak-eterm-yank-window** (*ID*) [Interactive Command]

Yank contents of an eterm window at point.

**emacspeak-toggle-eterm-autospeak** (**&optional** *PREFIX*) [Interactive Command]

Toggle state of eterm autospeak. When eterm autospeak is turned on and the terminal is in line mode, all output to the terminal is automatically spoken. Interactive prefix arg means toggle the global default value, and then set the current local value to the result.

## 11.23 emacspeak-filtertext

Automatically generated documentation for commands defined in module emacspeak-filtertext.

**emacspeak-filtertext** (*START END*) [Interactive Command]  
*control e ^*

Copy over text in region to special filtertext buffer in preparation for interactively filtering text.

**emacspeak-filtertext-mode** () [Interactive Command]

Major mode for FilterText interaction.

key binding — ——

= keep-lines ^ flush-lines r emacspeak-filtertext-revert

In addition to any hooks its parent mode ‘text-mode’ might have run, this mode runs the hook ‘emacspeak-filtertext-mode-hook’, as the final step during initialization.

**emacspeak-filtertext-revert** () [Interactive Command]

Revert to original text.

## 11.24 emacspeak-fix-interactive

Automatically generated documentation for commands defined in module emacspeak-fix-interactive.

**emacspeak-fix-all-recent-commands** () [Interactive Command]

Fix recently loaded interactive commands. This command looks through ‘load-history’ and fixes commands if necessary. Memoizes call in emacspeak-load-history-pointer to memoize this call.

`emacspeak-fix-commands-loaded-from` (*MODULE*) [Interactive Command]  
 Fix all commands loaded from a specified module.

## 11.25 emacspeak-forms

Automatically generated documentation for commands defined in module emacspeak-forms.

`emacspeak-forms-find-file` (*FILENAME*) [Interactive Command]  
 Visit a forms file

`emacspeak-forms-flush-unwanted-records` () [Interactive Command]  
 Prompt for pattern and flush matching lines

`emacspeak-forms-rerun-filter` () [Interactive Command]  
 Rerun filter –allows us to nuke more matching records

`emacspeak-forms-speak-field` () [Interactive Command]  
 Speak current form field name and value. Assumes that point is at the front of a field value.

`emacspeak-forms-summarize-current-position` () [Interactive Command]  
 Summarize current position in list of records

`emacspeak-forms-summarize-current-record` () [Interactive Command]  
 Summarize current record

## 11.26 emacspeak-gnus

Automatically generated documentation for commands defined in module emacspeak-gnus.

`emacspeak-gnus-summary-catchup-quietly-and-exit` () [Interactive Command]  
 Catch up on all articles in current group.

## 11.27 emacspeak-gomoku

Automatically generated documentation for commands defined in module emacspeak-gomoku.

`emacspeak-gomoku-display-statistics` () [Interactive Command]  
 Display statistics from previous games

`emacspeak-gomoku-goto-x-y` (*X Y*) [Interactive Command]  
 Prompt for and go to that square.

`emacspeak-gomoku-show-current-column` () [Interactive Command]  
 Aurally display current column

`emacspeak-gomoku-show-current-negative-diagonal` () [Interactive Command]  
 Aurally display current negative sloped diagonal

|                                                                                                                                       |                       |
|---------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-gomoku-show-current-positive-diagonal</code><br><code>()</code><br>Aurally display current positively sloped diagonal | [Interactive Command] |
| <code>emacspeak-gomoku-show-current-row</code> <code>()</code><br>Aurally display current row                                         | [Interactive Command] |
| <code>emacspeak-gomoku-speak-emacs-previous-move</code> <code>()</code><br>Speak emacs' previous move                                 | [Interactive Command] |
| <code>emacspeak-gomoku-speak-humans-previous-move</code> <code>()</code><br>Speak human' previous move                                | [Interactive Command] |
| <code>emacspeak-gomoku-speak-number-of-moves</code> <code>()</code><br>Speak number of moves so far                                   | [Interactive Command] |
| <code>emacspeak-gomoku-speak-square</code> <code>()</code><br>Speak coordinates and state of square at point                          | [Interactive Command] |

## 11.28 emacspeak-gridtext

Commentary: Emacspeak's table browsing mode allows one to efficiently access content that is tabular in nature. That module also provides functions for inferring table structure where possible. Often, such structure is hard to infer automatically –but might be known to the user e.g. treat columns 1 through 30 as one column of a table and so on. This module allows the user to specify a conceptual grid that is "overlaid" on the region of text to turn it into a table for tabular browsing. For now, elements of the grid are "one line" high –but that may change in the future if necessary. This module is useful for browsing structured text files and the output from programs that tabulate their output. It's also useful for handling multicolumn text.

Automatically generated documentation for commands defined in module emacspeak-gridtext.

|                                                                                                                        |                       |
|------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-gridtext-apply</code> ( <i>START END GRID</i> )<br><code>control e # a</code><br>Apply grid to region. | [Interactive Command] |
| <code>emacspeak-gridtext-load</code> ( <i>FILE</i> )<br><code>control e # l</code><br>Load saved grid settings.        | [Interactive Command] |
| <code>emacspeak-gridtext-save</code> ( <i>FILE</i> )<br><code>control e # s</code><br>Save out grid settings.          | [Interactive Command] |

## 11.29 emacspeak-hide

Commentary: Flexible hide and show for emacspeak. This module allows one to easily hide or expose blocks of lines starting with a common prefix. It is motivated by the need to flexibly hide quoted text in email but is designed to be more general. the prefix parsing is inspired by filladapt.el

Automatically generated documentation for commands defined in module emacspeak-hide.

**emacspeak-hide-or-expose-all-blocks** () [Interactive Command]  
Hide or expose all blocks in buffer.

**emacspeak-hide-or-expose-block** (&optional *PREFIX*) [Interactive Command]  
*control e j*

Hide or expose a block of text. This command either hides or exposes a block of text starting on the current line. A block of text is defined as a portion of the buffer in which all lines start with a common PREFIX. Optional interactive prefix arg causes all blocks in current buffer to be hidden or exposed.

**emacspeak-hide-speak-block-sans-prefix** () [Interactive Command]  
*control e control j*

Speaks current block after stripping its prefix. If the current block is not hidden, it first hides it. This is useful because as you locate blocks, you can invoke this command to listen to the block, and when you have heard enough navigate easily to move past the block.

## 11.30 emacspeak-ibuffer

Commentary: speech-enable ibuffer.el this is an alternative to buffer-menu

Automatically generated documentation for commands defined in module emacspeak-ibuffer.

**emacspeak-ibuffer-speak-buffer-line** () [Interactive Command]  
Speak information about this buffer

## 11.31 emacspeak-imenu

Automatically generated documentation for commands defined in module emacspeak-imenu.

**emacspeak-imenu-goto-next-index-position** () [Interactive Command]  
Goto the next index position in current buffer

**emacspeak-imenu-goto-previous-index-position** () [Interactive Command]  
Goto the previous index position in current buffer

**emacspeak-imenu-speak-this-section** () [Interactive Command]  
Speak upto start of next index entry

### 11.32 emacspeak-info

Automatically generated documentation for commands defined in module emacspeak-info.

**emacspeak-info-speak-header** () [Interactive Command]  
 Speak info header line.

**emacspeak-info-wizard** (*NODE-SPEC*) [Interactive Command]  
*control h TAB <f1> TAB <help> TAB*  
 Read a node spec from the minibuffer and launch Info-goto-node. See documentation for command 'Info-goto-node' for details on node-spec.

### 11.33 emacspeak-jabber

Commentary: emacs-jabber.el implements a jabber client for emacs emacs-jabber is hosted at sourceforge. I use emacs-jabber with my gmail.com account

Automatically generated documentation for commands defined in module emacspeak-jabber.

**emacspeak-jabber-popup-roster** () [Interactive Command]  
 Pop to Jabber roster.

### 11.34 emacspeak-keymap

Automatically generated documentation for commands defined in module emacspeak-keymap.

**emacspeak-keymap-choose-new-emacspeak-prefix** [Interactive Command]  
 (*PREFIX-KEY*)  
 Interactively select a new prefix key to use for all emacspeak commands. The default is to use 'C-e' This command lets you switch the prefix to something else. This is a useful thing to do if you run emacspeak on a remote machine from inside a terminal that is running inside a local emacspeak session. You can have the remote emacspeak use a different control key to give your fingers some relief.

### 11.35 emacspeak-loaddefs

Automatically generated documentation for commands defined in module emacspeak-loaddefs.

**emacspeak-mark-forward-mark** () [Interactive Command]  
*<control down>*  
 Jump to mark, and pop a new position for mark off the ring (does not affect global mark ring).  
 This function is advised.  
 After-advice 'emacspeak': Produce an auditory icon if possible.

**emacspeak-toggle-comint-output-monitor** (**&optional** [Interactive Command]  
*PREFIX*)  
*control e o*

Toggle state of Emacspeak comint monitor. When turned on, comint output is automatically spoken. Turn this on if you want your shell to speak its results. Interactive PREFIX arg means toggle the global default value, and then set the current local value to the result.

## 11.36 emacspeak-m-player

Commentary: Defines a simple derived mode for interacting with mplayer. mplayer is a versatile media player capable of playing many streaming formats and is especially useful for playing windows media (WMA) and streaming windows media (ASF) files. mplayer is available on the WWW: RPM package <http://mirrors.sctpc.com/dominik/linux/pkgs/mplayer/i586/mplayer-0.90pre5-2.i586.rpm> You may need the win32 codecs which can be downloaded from <http://ftp.lug.udel.edu/MPlayer/releases/w32codec-0.60.tar.bz2> Mplayer FAQ at <http://www.mplayerhq.hu/DOCS/faq.html> Mplayer docs at <http://www.mplayerhq.hu/DOCS/>

Automatically generated documentation for commands defined in module emacspeak-m-player.

- |                                                                                                                                   |                       |
|-----------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <b>emacspeak-m-player-add-equalizer</b> ()                                                                                        | [Interactive Command] |
| Add equalizer for next MPlayer invocation.                                                                                        |                       |
| Use arrow keys, page-up, page-down, home and end keys to manipulate the values. Hit enter to finish setting the equalizer values. |                       |
| The Mplayer equalizer provides 10 bands, G0 – G9, see the MPlayer man page for details.                                           |                       |
| <b>emacspeak-m-player-add-filter</b> ()                                                                                           | [Interactive Command] |
| Adds specified filter to use for the next invocation of MPlayer.                                                                  |                       |
| <b>emacspeak-m-player-alt-src-step</b> ( <i>STEP</i> )                                                                            | [Interactive Command] |
| Move within an ASF playlist.                                                                                                      |                       |
| <b>emacspeak-m-player-backward-10min</b> ()                                                                                       | [Interactive Command] |
| Move backward by ten minutes.                                                                                                     |                       |
| <b>emacspeak-m-player-backward-10s</b> ()                                                                                         | [Interactive Command] |
| Move back by 10 seconds.                                                                                                          |                       |
| <b>emacspeak-m-player-backward-1min</b> ()                                                                                        | [Interactive Command] |
| Move back by 1 minute.                                                                                                            |                       |
| <b>emacspeak-m-player-beginning-of-track</b> ()                                                                                   | [Interactive Command] |
| Move to beginning of track.                                                                                                       |                       |
| <b>emacspeak-m-player-command</b> ( <i>COMMAND-CHAR</i> )                                                                         | [Interactive Command] |
| Invoke MPlayer commands.                                                                                                          |                       |
| <b>emacspeak-m-player-customize-options</b> ()                                                                                    | [Interactive Command] |
| Use Customize to manipulate MPlayer options.                                                                                      |                       |

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                       |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-m-player-display-position ()</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | [Interactive Command] |
| Display current position in track and its length.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                       |
| <code>emacspeak-m-player-double-speed ()</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | [Interactive Command] |
| Scale speed by 2.0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                       |
| <code>emacspeak-m-player-end-of-track ()</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | [Interactive Command] |
| Move to beginning of track.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                       |
| <code>emacspeak-m-player-equalizer-control (V)</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | [Interactive Command] |
| Manipulate values in specified vector using minibuffer.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                       |
| <code>emacspeak-m-player-faster ()</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | [Interactive Command] |
| Speed up playback.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                       |
| <code>emacspeak-m-player-forward-10min ()</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | [Interactive Command] |
| Move forward by ten minutes.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                       |
| <code>emacspeak-m-player-forward-10s ()</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | [Interactive Command] |
| Move forward by 10 seconds.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                       |
| <code>emacspeak-m-player-forward-1min ()</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | [Interactive Command] |
| Move forward by 1 minute.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                       |
| <code>emacspeak-m-player-half-speed ()</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | [Interactive Command] |
| Scale speed by 0.5.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                       |
| <code>emacspeak-m-player-load-file (F)</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | [Interactive Command] |
| Load specified file.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                       |
| <code>emacspeak-m-player-load-playlist (F)</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | [Interactive Command] |
| Load specified playlist file.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                       |
| <code>emacspeak-m-player-mode ()</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | [Interactive Command] |
| Major mode for m-player interaction.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                       |
| key binding — ——                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                       |
| RET emacspeak-m-player SPC emacspeak-m-player-pause + emacspeak-m-player-volume-up , emacspeak-m-player-backward-10s - emacspeak-m-player-volume-down . emacspeak-m-player-forward-10s : emacspeak-m-player < emacspeak-m-player-backward-1min = emacspeak-m-player-volume-up > emacspeak-m-player-forward-1min ? emacspeak-m-player-display-position L emacspeak-m-player-load-playlist O emacspeak-m-player-reset-options [ emacspeak-m-player-slower ] emacspeak-m-player-faster b bury-buffer e emacspeak-m-player-add-equalizer f emacspeak-m-player-add-filter g emacspeak-m-player-seek-absolute l emacspeak-m-player-load-file n emacspeak-m-player-next-track o emacspeak-m-player-customize-options p emacspeak-m-player-previous-track q emacspeak-m-player-quit r emacspeak-m-player-seek-relative s emacspeak-m-player-scale-speed t emacspeak-m-player-play-tree-step { emacspeak-m-player-half-speed } emacspeak-m-player-double-speed DEL emacspeak-m-player-reset-speed <down> emacspeak-m-player-forward-1min <end> emacspeak-m-player-end-of-track <home> emacspeak-m-player-beginning-of-track |                       |

<left> emacspeak-m-player-backward-10s <next> emacspeak-m-player-forward-10min  
 <prior> emacspeak-m-player-backward-10min <right> emacspeak-m-player-forward-10s  
 <up> emacspeak-m-player-backward-1min

In addition to any hooks its parent mode ‘comint-mode’ might have run, this mode runs the hook ‘emacspeak-m-player-mode-hook’, as the final step during initialization.

|                                                          |                       |
|----------------------------------------------------------|-----------------------|
| <code>emacspeak-m-player-next-track ()</code>            | [Interactive Command] |
| Move to next track.                                      |                       |
| <code>emacspeak-m-player-pause ()</code>                 | [Interactive Command] |
| Pause or unpause media player.                           |                       |
| <code>emacspeak-m-player-play-tree-step (STEP)</code>    | [Interactive Command] |
| Move within the play tree.                               |                       |
| <code>emacspeak-m-player-play-tree-up (STEP)</code>      | [Interactive Command] |
| Move within the play tree.                               |                       |
| <code>emacspeak-m-player-previous-track ()</code>        | [Interactive Command] |
| Move to previous track.                                  |                       |
| <code>emacspeak-m-player-quit ()</code>                  | [Interactive Command] |
| Quit media player.                                       |                       |
| <code>emacspeak-m-player-reset-options ()</code>         | [Interactive Command] |
| Reset MPlayer options to initial defaults.               |                       |
| <code>emacspeak-m-player-reset-speed ()</code>           | [Interactive Command] |
| Reset playing speed to normal.                           |                       |
| <code>emacspeak-m-player-scale-speed (FACTOR)</code>     | [Interactive Command] |
| Scale speed by specified factor.                         |                       |
| <code>emacspeak-m-player-seek-absolute (POSITION)</code> | [Interactive Command] |
| Seek to absolute specified position.                     |                       |
| <code>emacspeak-m-player-seek-relative (OFFSET)</code>   | [Interactive Command] |
| Seek by offset into stream from current position.        |                       |
| <code>emacspeak-m-player-slower ()</code>                | [Interactive Command] |
| Slow down playback.                                      |                       |
| <code>emacspeak-m-player-volume-down ()</code>           | [Interactive Command] |
| Decrease volume.                                         |                       |
| <code>emacspeak-m-player-volume-up ()</code>             | [Interactive Command] |
| Increase volume.                                         |                       |
| <code>emacspeak-multimedia ()</code>                     | [Interactive Command] |
| <code>control e ;</code>                                 |                       |
| Start or control Emacspeak multimedia player.            |                       |

### 11.37 emacspeak-madplay

Commentary: Defines a simple derived mode for interacting with madplay. madplay navigation commands work via single keystrokes.

Automatically generated documentation for commands defined in module emacspeak-madplay.

**emacspeak-madplay** (*RESOURCE*) [Interactive Command]  
 Play specified resource using madplay. Resource is an MP3 file or directory containing mp3 files. The player is placed in a buffer in emacspeak-madplay-mode.

**emacspeak-madplay-madplay-call-command** () [Interactive Command]  
 Call appropriate madplay command.

**emacspeak-madplay-madplay-command** (*CHAR*) [Interactive Command]  
 Execute Madplay command.

**emacspeak-madplay-mode** () [Interactive Command]  
 Major mode for madplay interaction.

key binding — ——

+ emacspeak-madplay-madplay-call-command - emacspeak-madplay-madplay-call-command b emacspeak-madplay-madplay-call-command f emacspeak-madplay-madplay-call-command o emacspeak-madplay p .. q emacspeak-madplay-madplay-call-command s .. t emacspeak-madplay-madplay-call-command <left> emacspeak-aumix-wave-decrease <right> emacspeak-aumix-wave-increase

This mode runs the hook ‘emacspeak-madplay-mode-hook’, as the final step during initialization.

### 11.38 emacspeak-man

Automatically generated documentation for commands defined in module emacspeak-man.

**emacspeak-man-browse-man-page** () [Interactive Command]  
 Browse the man page –read it a paragraph at a time

**emacspeak-man-speak-this-section** () [Interactive Command]  
 Speak current section

### 11.39 emacspeak-ocr

Commentary: This module defines Emacspeak front-end to OCR. This module assumes that sane is installed and working for image acquisition, and that there is an OCR engine that can take acquired images and produce text. Prerequisites: Sane installed and working. scanimage to generate tiff files from scanner. tiffcp to compress the tiff file. working ocr executable by default this module assumes that the OCR executable is named "ocr"

Automatically generated documentation for commands defined in module emacspeak-ocr.

**emacspeak-ocr** () [Interactive Command]  
*control e control o*

An OCR front-end for the Emacspeak desktop.

Page image is acquired using tools from the SANE package. The acquired image is run through the OCR engine if one is available, and the results placed in a buffer that is suitable for browsing the results.

For detailed help, invoke command `emacspeak-ocr` bound to C-e C-o to launch `emacspeak-ocr-mode`, and press '?' to display mode-specific help for `emacspeak-ocr-mode`.

**emacspeak-ocr-backward-page** (&optional [Interactive Command]  
*COUNT-IGNORED*)

Like backward page, but tracks page number of current document.

**emacspeak-ocr-customize** () [Interactive Command]  
 Customize OCR settings.

**emacspeak-ocr-forward-page** (&optional [Interactive Command]  
*COUNT-IGNORED*)

Like forward page, but tracks page number of current document.

**emacspeak-ocr-mode** () [Interactive Command]

An OCR front-end for the Emacspeak desktop.

Pre-requisites:

- 1) A working scanner back-end like SANE on Linux.
- 2) An OCR engine.

1: Make sure your scanner back-end works, and that you have the utilities to scan a document and acquire an image as a tiff file. Then set variable `emacspeak-ocr-scan-image-program` to point at this utility. By default, this is set to 'scanimage' which is the image scanning utility provided by SANE.

By default, this front-end attempts to compress the acquired tiff image; make sure you have a utility like `tiffcp`. Variable `emacspeak-ocr-compress-image` is set to 'tiffcp' by default; if you use something else, you should customize this variable.

2: Next, make sure you have an OCR engine installed and working. By default this front-end assumes that OCR is available as `/usr/bin/ocr`.

Once you have ensured that acquiring an image and applying OCR to it work independently of Emacs, you can use this Emacspeak front-end to enable easy OCR access from within Emacspeak.

The Emacspeak OCR front-end is launched by command `emacspeak-ocr` bound to C-e C-o.

This command switches to a special buffer that has OCR commands bounds to single keystrokes— see the `ke-binding` list at the end of this description. Use Emacs online help facility to look up help on these commands.

`emacspeak-ocr-mode` provides the necessary functionality to scan, OCR, read and save documents. By default, scanned images and the resulting text are saved under

directory `~/ocr`; see variable `emacspeak-ocr-working-directory`. Invoking command `emacspeak-ocr-open-working-directory` bound to `M-x emacspeak-ocr-open-working-directory` will open this directory.

By default, the document being scanned is named ‘untitled’. You can name the document by using command `emacspeak-ocr-name-document` bound to `M-x emacspeak-ocr-name-document`. The document name is used in constructing the name of the image and text files.

Key Bindings:

See key binding — ——

`RET` `emacspeak-ocr-scan-and-recognize` `SPC` `emacspeak-ocr-read-current-page` `1`  
`.. 9` `emacspeak-ocr-page` `?` `describe-mode` `C` `emacspeak-ocr-set-compress-image-`  
`options` `I` `emacspeak-ocr-set-scan-image-options` `[` `emacspeak-ocr-backward-page`  
`]` `emacspeak-ocr-forward-page` `c` `emacspeak-ocr-customize` `d` `emacspeak-ocr-`  
`open-working-directory` `i` `emacspeak-ocr-scan-image` `j` `emacspeak-ocr-scan-photo`  
`n` `emacspeak-ocr-name-document` `o` `emacspeak-ocr-recognize-image` `p`  
`emacspeak-ocr-page` `q` `bury-buffer` `s` `emacspeak-ocr-save-current-page` `w`  
`emacspeak-ocr-write-document`

In addition to any hooks its parent mode ‘text-mode’ might have run, this mode runs the hook ‘`emacspeak-ocr-mode-hook`’, as the final step during initialization.

|                                                                                                                                                                                       |                       |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-ocr-name-document</code> ( <i>NAME</i> )                                                                                                                              | [Interactive Command] |
| Name document being scanned in the current OCR buffer. Pick a short but meaningful name.                                                                                              |                       |
| <code>emacspeak-ocr-open-working-directory</code> ()                                                                                                                                  | [Interactive Command] |
| Launch <code>dired</code> on OCR working directory.                                                                                                                                   |                       |
| <code>emacspeak-ocr-page</code> ()                                                                                                                                                    | [Interactive Command] |
| Move to specified page.                                                                                                                                                               |                       |
| <code>emacspeak-ocr-read-current-page</code> ()                                                                                                                                       | [Interactive Command] |
| Speaks current page.                                                                                                                                                                  |                       |
| <code>emacspeak-ocr-recognize-image</code> ()                                                                                                                                         | [Interactive Command] |
| Run OCR engine on current image. Prompts for image file if file corresponding to the expected ‘current page’ is not found.                                                            |                       |
| <code>emacspeak-ocr-save-current-page</code> ()                                                                                                                                       | [Interactive Command] |
| Writes out recognized text from current page to an appropriately named file.                                                                                                          |                       |
| <code>emacspeak-ocr-scan-and-recognize</code> ()                                                                                                                                      | [Interactive Command] |
| Scan in a page and run OCR engine on it. Use this command once you’ve verified that the separate steps of acquiring an image and running the OCR engine work correctly by themselves. |                       |
| <code>emacspeak-ocr-scan-image</code> ()                                                                                                                                              | [Interactive Command] |
| Acquire page image.                                                                                                                                                                   |                       |

**emacspeak-ocr-scan-photo** (**&optional** *METADATA*) [Interactive Command]  
Scan in a photograph. The scanned image is converted to JPEG.

**emacspeak-ocr-set-compress-image-options** [Interactive Command]  
(*SETTING*)  
Interactively update image compression options. Prompts with current setting in the minibuffer. Setting persists for current Emacs session.

**emacspeak-ocr-set-scan-image-options** (*SETTING*) [Interactive Command]  
Interactively update scan image options. Prompts with current setting in the minibuffer. Setting persists for current Emacs session.

**emacspeak-ocr-write-document** () [Interactive Command]  
Writes out recognized text from all pages in current document.

## 11.40 emacspeak-outline

Commentary: Provide additional advice to outline-mode

Automatically generated documentation for commands defined in module emacspeak-outline.

**emacspeak-outline-speak-backward-heading** () [Interactive Command]  
Analogous to outline-backward-same-level except that the outline section is optionally spoken

**emacspeak-outline-speak-forward-heading** () [Interactive Command]  
Analogous to outline-forward-same-level, except that the outline section is optionally spoken

**emacspeak-outline-speak-next-heading** () [Interactive Command]  
Analogous to outline-next-visible-heading, except that the outline section is optionally spoken

**emacspeak-outline-speak-previous-heading** () [Interactive Command]  
Analogous to outline-previous-visible-heading, except that the outline section is optionally spoken

**emacspeak-outline-speak-this-heading** () [Interactive Command]  
Speak current outline section starting from point

## 11.41 emacspeak-pronounce

Commentary: This module implements user customizable pronunciation dictionaries for emacspeak. Custom pronunciations can be defined per file, per directory and/or per major mode. Emacspeak maintains a persistent user dictionary upon request and loads these in new emacspeak sessions. This module implements the user interface to the custom dictionary as well as providing the internal API used by the rest of emacspeak in using the dictionary. Algorithm:

The persistent dictionary is a hash table where the hash keys are filenames, directory names, or major-mode names. The hash values are association lists defining the dictionary. Users of this module can retrieve a dictionary made up of all applicable association lists for a given file.

Automatically generated documentation for commands defined in module `emacspeak-pronounce`.

`emacspeak-pronounce-clear-dictionaries` () [Interactive Command]  
Clear all current pronunciation dictionaries.

`emacspeak-pronounce-define-local-pronunciation` [Interactive Command]  
(*WORD PRONUNCIATION*)  
Define buffer local pronunciation. Argument *WORD* specifies the word which should be pronounced as specified by *PRONUNCIATION*.

`emacspeak-pronounce-define-pronunciation` () [Interactive Command]  
Interactively define entries in the pronunciation dictionaries. Default term to define is delimited by region. First loads any persistent dictionaries if not already loaded.

`emacspeak-pronounce-define-template-pronunciation` [Interactive Command]  
( )  
Interactively define template entries in the pronunciation dictionaries. Default term to define is delimited by region. First loads any persistent dictionaries if not already loaded.

`emacspeak-pronounce-dispatch` () [Interactive Command]  
*control e meta d*  
Provides the user interface front-end to Emacspeak's pronunciation dictionaries.

`emacspeak-pronounce-edit-pronunciations` (*KEY*) [Interactive Command]  
Prompt for and launch a pronunciation editor on the specified pronunciation dictionary key.

`emacspeak-pronounce-load-dictionaries` (&optional *FILENAME*) [Interactive Command]  
Load pronunciation dictionaries. Optional argument *FILENAME* specifies the dictionary file.

`emacspeak-pronounce-refresh-pronunciations` () [Interactive Command]  
Refresh pronunciation table for current buffer. Activates pronunciation dictionaries if not already active.

`emacspeak-pronounce-save-dictionaries` () [Interactive Command]  
Writes out the persistent emacspeak pronunciation dictionaries.

`emacspeak-pronounce-toggle-use-of-dictionaries` [Interactive Command]  
(&optional *STATE*)  
Toggle use of pronunciation dictionaries in current buffer. Pronunciations can be defined on a per file, per directory and/or per mode basis. Pronunciations are activated on a per buffer basis. Turning on the use of pronunciation dictionaries results

in emacspeak composing a pronunciation table based on the currently defined pronunciation dictionaries. After this, the pronunciations will be applied whenever text in the buffer is spoken. Optional argument state can be used from Lisp programs to explicitly turn pronunciations on or off.

**emacspeak-pronounce-yank-word ()** [Interactive Command]  
Yank word at point into minibuffer.

## 11.42 emacspeak-python

Commentary: This speech-enables python-mode available on sourceforge

Automatically generated documentation for commands defined in module emacspeak-python.

**emacspeak-py-next-block ()** [Interactive Command]  
Move forward to the beginning of the next block.

**emacspeak-py-previous-block ()** [Interactive Command]  
Move backward to the beginning of the current block. If already at the beginning then move to previous block.

## 11.43 emacspeak-realaudio

Commentary: Assuming you have a correctly configured RealAudio player, this package provides single click access to starting and stopping a RealAudio stream from anywhere on the Emacspeak desktop. Before using this package, make sure that your realaudio player works outside Emacs. Then set variable Emacspeak-realaudio-player to point to the program you use to play RealAudio streams.

Automatically generated documentation for commands defined in module emacspeak-realaudio.

**emacspeak-realaudio-browse (*RAMFILE* &optional *START-TIME*)** [Interactive Command]  
Browse RAM file before playing the selected component.

**emacspeak-realaudio-get-current-time-in-seconds ()** [Interactive Command]  
Return current time in seconds.

**emacspeak-realaudio-mode ()** [Interactive Command]  
Major mode for streaming audio.  
key binding — ——  
This mode runs the hook ‘emacspeak-realaudio-mode-hook’, as the final step during initialization.

**emacspeak-realaudio-play (*RESOURCE* &optional *PLAY-LIST NOSELECT*)** [Interactive Command]  
Play a realaudio stream. Uses files from your Realaudio shortcuts directory for completion. See documentation for user configurable variable emacspeak-realaudio-shortcuts-directory.

|                                                                                                                   |                       |
|-------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-realaudio-select-realaudio-buffer ()</code>                                                       | [Interactive Command] |
| Switch to realaudio buffer.                                                                                       |                       |
| <code>emacspeak-realaudio-set-end-mark (&amp;optional MARK-TIME)</code>                                           | [Interactive Command] |
| Set end mark. Default is to set marker to current play time. Mark is specified in seconds.                        |                       |
| <code>emacspeak-realaudio-set-start-mark (&amp;optional MARK-TIME)</code>                                         | [Interactive Command] |
| Set start mark. Default is to set marker to current play time. Mark is specified in seconds.                      |                       |
| <code>emacspeak-realaudio-stop ()</code>                                                                          | [Interactive Command] |
| Stop playing realaudio                                                                                            |                       |
| <code>emacspeak-realaudio-trplayer-call-command ()</code>                                                         | [Interactive Command] |
| Call appropriate TRPlayer command.                                                                                |                       |
| <code>emacspeak-realaudio-trplayer-command (CHAR)</code>                                                          | [Interactive Command] |
| Execute TRPlayer command.                                                                                         |                       |
| <code>emacspeak-realaudio-write-mp3-clip (START END FILE)</code>                                                  | [Interactive Command] |
| Writes specified clip from current mp3 stream. Prompts for start and end times as well as file to save the clippi |                       |

## 11.44 emacspeak-redefine

Automatically generated documentation for commands defined in module emacspeak-redefine.

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                       |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-backward-char (&amp;optional ARG)</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | [Interactive Command] |
| <i>&lt;left&gt; control b</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                       |
| Backward-char redefined to speak char moved to.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                       |
| <code>emacspeak-forward-char (&amp;optional ARG)</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | [Interactive Command] |
| <i>&lt;right&gt; control f</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                       |
| Forward-char redefined to speak char moved to.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                       |
| <code>emacspeak-self-insert-command (&amp;optional ARG)</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | [Interactive Command] |
| <i>Generic char 491520: all of JISX0213-2 Generic char 458752: all of Tibetan 2-col Generic char 442368: all of Indian 2-col Generic char 425984: all of CNS11643-7 Generic char 409600: all of CNS11643-6 Generic char 393216: all of CNS11643-5 Generic char 376832: all of CNS11643-4 Generic char 360448: all of CNS11643-3 Generic char 344064: all of Ethiopic Generic char 327680: all of Unicode subset Generic char 311296: all of Unicode subset 3 Generic char 294912: all of Unicode subset 2 Generic char 278528: all of Tibetan 1-col Generic char 262144: all of Indian glyph Generic char 14464: all of IS 13194 Generic char 14336: all of Arabic 2-col Generic char</i> |                       |

7040: all of Lao Generic char 6912: all of rev ASCII Generic char 6784: all of Arabic 1-col Generic char 6656: all of Arabic digit Generic char 6528: all of VISCII upper Generic char 6400: all of VISCII lower Generic char 6272: all of IPA Generic char 6144: all of SiSheng Generic char 16384: all of Big5 (Level-2) Generic char 147456: all of Big5 (Level-1) Generic char 131072: all of JISX0213-1 Generic char 114688: all of CNS11643-2 Generic char 98304: all of CNS11643-1 Generic char 81920: all of JISX0212 Generic char 65536: all of KSC5601 Generic char 49152: all of JISX0208 Generic char 32768: all of GB2312 Generic char 16384: all of JISX0208.1978 Generic char 3968: all of RHP of Latin-8 Generic char 3840: all of RHP of Latin-9 Generic char 3712: all of RHP of Latin-5 Generic char 3584: all of RHP of ISO8859/5 Generic char 3328: all of JISX0201 Roman Generic char 3200: all of JISX0201 Katakana Generic char 3072: all of RHP of ISO8859/8 Generic char 2944: all of RHP of ISO8859/6 Generic char 2816: all of RHP of ISO8859/7 Generic char 2688: all of RHP of TIS620 Generic char 2560: all of RHP of Latin-4 Generic char 2432: all of RHP of Latin-3 Generic char 2304: all of RHP of Latin-2 Generic char 2176: all of RHP of Latin-1  
 ~ } | { z y x w v u t s r q p o n m l k j i h g f e d c b a ' \_ ^ ] \ [ cap Z cap Y cap X cap W cap V cap U cap T cap S cap R cap Q cap P cap O cap N cap M cap L cap K cap J cap I cap H cap G cap F cap E cap D cap C cap B cap A @ ? > = < ; : 9 8 7 6 5 4 3 2 1 0 / . - , + \* ) ( ' & % \$ # ! SPACE

Insert a character. Speaks the character if emacspeak-character-echo is true. See command emacspeak-toggle-word-echo bound to C-e d w. Toggle variable dtk-stop-immediately-while-typing if you want to have speech flush as you type.

## 11.45 emacspeak-remote

Automatically generated documentation for commands defined in module emacspeak-remote.

**emacspeak-remote-connect-to-server** (*HOST PORT*) [Interactive Command]  
*control e meta r*

Connect to and start using remote speech server running on host *host* and listening on port *port*. Host is the hostname of the remote server, typically the desktop machine. Port is the tcp port that that host is listening on for speech requests.

**emacspeak-remote-home** () [Interactive Command]  
 Open ssh session to where we came from. Uses value returned by ‘emacspeak-remote-get-current-remote-hostname’.

**emacspeak-remote-quick-connect-to-server** () [Interactive Command]  
 Connect to remote server. Does not prompt for host or port, but quietly uses the guesses that appear as defaults when prompting. Use this once you are sure the guesses are usually correct.

**emacspeak-remote-ssh-to-server** (*LOGIN HOST PORT*) [Interactive Command]  
 Open ssh session to where we came from.

## 11.46 emacspeak-rmail

Automatically generated documentation for commands defined in module emacspeak-rmail.

`emacspeak-rmail-speak-current-message-labels ()` [Interactive Command]  
Speak labels of current message

`emacspeak-rmail-summarize-current-message ()` [Interactive Command]  
Summarize current message

## 11.47 emacspeak-solitaire

Automatically generated documentation for commands defined in module emacspeak-solitaire.

`emacspeak-solitaire-show-column ()` [Interactive Command]  
Display current row auditorally

`emacspeak-solitaire-show-row ()` [Interactive Command]  
Display current row auditorally

`emacspeak-solitaire-speak-coordinates ()` [Interactive Command]  
Speak coordinates of current position

## 11.48 emacspeak-sounds

Commentary: This module provides the interface for generating auditory icons in emacspeak. Design goal: 1) Auditory icons should be used to provide additional feedback, not as a gimmick. 2) The interface should be usable at all times without the icons: e.g. when on a machine without a sound card. 3) General principle for when to use an icon: Convey information about events taking place in parallel. For instance, if making a selection automatically moves the current focus to the next choice, We speak the next choice, while indicating the fact that something was selected with a sound cue. This interface will assume the availability of a shell command "play" that can take one or more sound files and play them. This module will also provide a mapping between names in the elisp world and actual sound files. Modules that wish to use auditory icons should use these names, instead of actual file names. As of Emacspeak 13.0, this module defines a themes architecture for auditory icons. Sound files corresponding to a given theme are found in appropriate subdirectories of emacspeak-sounds-directory

Automatically generated documentation for commands defined in module emacspeak-sounds.

`emacspeak-play-all-icons ()` [Interactive Command]  
Plays all defined icons and speaks their names.

`emacspeak-set-auditory-icon-player (PLAYER)` [Interactive Command]  
*control e meta a*

Select player used for producing auditory icons. Recommended choices:

emacspeak-serve-auditory-icon for the wave device. emacspeak-queue-auditory-icon when using software TTS.

- emacspeak-sounds-reset-local-player** () [Interactive Command]  
 Ask Emacspeak to use a local audio player. This lets me have Emacspeak switch to using audioplay on solaris after I've used it for a while from a remote session where it would use the more primitive speech-server based audio player.
- emacspeak-sounds-reset-sound** () [Interactive Command]  
 Reload sound drivers.
- emacspeak-sounds-select-theme** (*THEME*) [Interactive Command]  
*control e* )  
 Select theme for auditory icons.
- emacspeak-toggle-auditory-icons** (&optional *PREFIX*) [Interactive Command]  
*control e control a*  
 Toggle use of auditory icons. Optional interactive PREFIX arg toggles global value.

## 11.49 emacspeak-speak

Commentary: This module defines the core speech services used by emacspeak. It depends on the speech server interface modules It protects other parts of emacspeak from becoming dependent on the speech server modules

Automatically generated documentation for commands defined in module emacspeak-speak.

- emacspeak-audio-annotate-paragraphs** () [Interactive Command]  
 Set property auditory-icon at front of all paragraphs.
- emacspeak-blink-matching-open** () [Interactive Command]  
 Move cursor momentarily to the beginning of the sexp before point. Also display match context in minibuffer.
- emacspeak-completion-pick-completion** () [Interactive Command]  
 Pick completion and return safely where we came from.
- emacspeak-completions-move-to-completion-group** () [Interactive Command]  
 Move to group of choices beginning with character last typed. If no such group exists, then we try to search for that char, or dont move.
- emacspeak-dial-dtk** (*NUMBER*) [Interactive Command]  
*control e d t*  
 Prompt for and dial a phone NUMBER with the Dectalk.
- emacspeak-execute-repeatedly** (*COMMAND*) [Interactive Command]  
 Execute COMMAND repeatedly.
- emacspeak-mark-backward-mark** () [Interactive Command]  
*<control up>*  
 Cycle backward through the mark ring.

- `emacspeak-owindow-next-line` (*COUNT*) [Interactive Command]  
*ESCAPE* <*down*>  
 Move to the next line in the other window and speak it. Numeric prefix arg  
*COUNT* can specify number of lines to move.
- `emacspeak-owindow-previous-line` (*COUNT*) [Interactive Command]  
*ESCAPE* <*up*>  
 Move to the next line in the other window and speak it. Numeric prefix arg  
*COUNT* specifies number of lines to move.
- `emacspeak-owindow-scroll-down` () [Interactive Command]  
*ESCAPE* <*prior*>  
 Scroll down the window that command ‘other-window’ would move to. Speak the  
 window contents after scrolling.
- `emacspeak-owindow-scroll-up` () [Interactive Command]  
*ESCAPE* <*next*>  
 Scroll up the window that command ‘other-window’ would move to. Speak the win-  
 dow contents after scrolling.
- `emacspeak-owindow-speak-line` () [Interactive Command]  
*ESCAPE* <*select*>  
 Speak the current line in the other window.
- `emacspeak-read-next-line` (&**optional** *ARG*) [Interactive Command]  
*control e* <*down*>  
 Read next line, specified by an offset, without moving. Default is to read the next  
 line.
- `emacspeak-read-next-word` (&**optional** *ARG*) [Interactive Command]  
 Read next word, specified as a numeric arg, without moving. Default is to read the  
 next word.
- `emacspeak-read-previous-line` (&**optional** *ARG*) [Interactive Command]  
*control e* <*up*>  
 Read previous line, specified by an offset, without moving. Default is to read the  
 previous line.
- `emacspeak-read-previous-word` (&**optional** *ARG*) [Interactive Command]  
 Read previous word, specified as a prefix arg, without moving. Default is to read  
 the previous word.
- `emacspeak-speak-and-skip-extent-upto-char` (*CHAR*) [Interactive Command]  
 Search forward from point until we hit char. Speak text between point and the char  
 we hit.
- `emacspeak-speak-and-skip-extent-upto-this-char` () [Interactive Command]  
 Speak extent delimited by point and last character typed.

**emacspeak-speak-browse-buffer** (**&optional** *BROWSE*) [Interactive Command]  
*control e ,*

Browse current buffer. Default is to speak chunk having current personality. Interactive prefix arg ‘browse’ repeatedly browses through chunks having same personality as the current text chunk.

**emacspeak-speak-buffer** (**&optional** *ARG*) [Interactive Command]  
*control e b*

Speak current buffer contents. With prefix *ARG*, speaks the rest of the buffer from point. Negative prefix arg speaks from start of buffer to point. If voice lock mode is on, the paragraphs in the buffer are voice annotated first, see command ‘emacspeak-speak-voice-annotate-paragraphs’.

**emacspeak-speak-buffer-filename** (**&optional** *FILENAME*) [Interactive Command]  
*control e f*

Speak name of file being visited in current buffer. Speak default directory if invoked in a dired buffer, or when the buffer is not visiting any file. Interactive prefix arg ‘filename’ speaks only the final path component. The result is put in the kill ring for convenience.

**emacspeak-speak-buffer-interactively** () [Interactive Command]  
*control e cap B*

Speak the start of, rest of, or the entire buffer. ‘s’ to speak the start. ‘r’ to speak the rest. any other key to speak entire buffer.

**emacspeak-speak-char** (**&optional** *PREFIX*) [Interactive Command]  
*control e c*

Speak character under point. Pronounces character phonetically unless called with a *PREFIX* arg.

**emacspeak-speak-comint-send-input** () [Interactive Command]  
 Causes output to be spoken i.e., as if comint autospeak were turned on.

**emacspeak-speak-continuously** () [Interactive Command]  
*control e RETURN*

Speak a buffer continuously. First prompts using the minibuffer for the kind of action to perform after speaking each chunk. E.G. speak a line at a time etc. Speaking commences at current buffer position. Pressing C-g breaks out, leaving point on last chunk that was spoken. Any other key continues to speak the buffer.

**emacspeak-speak-current-column** () [Interactive Command]  
*control e =*

Speak the current column.

**emacspeak-speak-current-field** () [Interactive Command]  
*control e .*

Speak current field.

- emacspeak-speak-current-kill** (*COUNT*) [Interactive Command]  
*control e k*  
 Speak the current kill entry. This is the text that will be yanked in by the next C-y. Prefix numeric arg, COUNT, specifies that the text that will be yanked as a result of a C-y followed by count-1 M-y be spoken. The kill number that is spoken says what numeric prefix arg to give to command yank.
- emacspeak-speak-current-mark** (*COUNT*) [Interactive Command]  
*control e control @*  
 Speak the line containing the mark. With no argument, speaks the line containing the mark—this is where ‘exchange-point-and-mark’ C-x C-x would jump. Numeric prefix arg ‘COUNT’ speaks line containing mark ‘n’ where ‘n’ is one less than the number of times one has to jump using ‘set-mark-command’ to get to this marked position. The location of the mark is indicated by an aural highlight achieved by a change in voice personality.
- emacspeak-speak-current-percentage** () [Interactive Command]  
*control e %*  
 Announce the percentage into the current buffer.
- emacspeak-speak-current-window** () [Interactive Command]  
 Speak contents of current window. Speaks entire window irrespective of point.
- emacspeak-speak-display-char** (&optional *PREFIX*) [Interactive Command]  
 Display char under point using current speech display table. Behavior is the same as command ‘emacspeak-speak-char’ bound to C-e c for characters in the range 0–127. Optional argument PREFIX specifies that the character should be spoken phonetically.
- emacspeak-speak-front-of-buffer** () [Interactive Command]  
 Speak the buffer from start to point
- emacspeak-speak-help** (&optional *ARG*) [Interactive Command]  
*control e h*  
 Speak help buffer if one present. With prefix arg, speaks the rest of the buffer from point. Negative prefix arg speaks from start of buffer to point.
- emacspeak-speak-help-interactively** () [Interactive Command]  
 Speak the start of, rest of, or the entire help. ‘s’ to speak the start. ‘r’ to speak the rest. any other key to speak entire help.
- emacspeak-speak-line** (&optional *ARG*) [Interactive Command]  
*control e l*  
 Speaks current line. With prefix ARG, speaks the rest of the line from point. Negative prefix optional arg speaks from start of line to point. Voicifies if option ‘voice-lock-mode’ is on. Indicates indentation with a tone if audio indentation is in use. Indicates position of point with an aural highlight if option ‘emacspeak-show-point’ is turned on—see command ‘emacspeak-show-point’ bound to M-x emacspeak-show-point. Lines that start hidden blocks of text, e.g. outline header lines, or header

lines of blocks created by command ‘emacspeak-hide-or-expose-block’ are indicated with auditory icon ellipses.

**emacspeak-speak-line-interactively** () [Interactive Command]  
*control e cap L*

Speak the start of, rest of, or the entire line. ‘s’ to speak the start. ‘r’ to speak the rest. any other key to speak entire line.

**emacspeak-speak-line-number** () [Interactive Command]  
*control e control l*

Print the current buffer line number and narrowed line number of point.

**emacspeak-speak-line-set-column-filter** (*FILTER*) [Interactive Command]  
*control e |*

Set up filter for selectively speaking or ignoring portions of lines. The filter is specified as a list of pairs. For example, to filter columns 1 – 10 and 20 – 25, specify filter as ((0 9) (20 25)). Filter settings are persisted across sessions. A persisted filter is used as the default when prompting for a filter. This allows one to accumulate a set of filters for specific files like /var/adm/messages and /var/adm/maillog over time. Option emacspeak-speak-line-invert-filter determines the sense of the filter.

**emacspeak-speak-load-directory-settings** [Interactive Command]  
 (&optional *DIRECTORY*)

Load a directory specific Emacspeak settings file. This is typically used to load up settings that are specific to an electronic book consisting of many files in the same directory.

**emacspeak-speak-message-again** (&optional [Interactive Command]  
*FROM-MESSAGE-CACHE*)

*control e a*

Speak the last message from Emacs once again. Optional interactive prefix arg ‘from-message-cache’ speaks message cached from the most recent call to function ‘message’. The message is also placed in the kill ring for convenient yanking if ‘emacspeak-speak-message-again-should-copy-to-kill-ring’ is set.

**emacspeak-speak-message-at-time** (*TIME MESSAGE*) [Interactive Command]  
*control e @*

Set up ring-at-time to speak message at specified time. Provides simple stop watch functionality in addition to other things. See documentation for command run-at-time for details on time-spec.

**emacspeak-speak-minibuffer** (&optional *ARG*) [Interactive Command]

Speak the minibuffer contents With prefix arg, speaks the rest of the buffer from point. Negative prefix arg speaks from start of buffer to point.

**emacspeak-speak-minor-mode-line** () [Interactive Command]  
*control e cap M*

Speak the minor mode-information.

- emacspeak-speak-mode-line** (**&optional** *BUFFER-INFO*) [Interactive Command]  
*control e m*  
 Speak the mode-line. Interactive prefix arg speaks buffer info.
- emacspeak-speak-next-field** () [Interactive Command]  
*control e >*  
 Move to and speak next field.
- emacspeak-speak-next-personality-chunk** () [Interactive Command]  
 Moves to the front of next chunk having current personality. Speak that chunk after moving.
- emacspeak-speak-next-window** () [Interactive Command]  
*control e control n*  
 Speak the next window.
- emacspeak-speak-other-buffer** (*BUFFER*) [Interactive Command]  
*control e meta b*  
 Speak specified buffer. Useful to listen to a buffer without switching contexts.
- emacspeak-speak-other-window** (**&optional** *ARG*) [Interactive Command]  
 Speak contents of ‘other’ window. Speaks entire window irrespective of point. Semantics of ‘other’ is the same as for the builtin Emacs command ‘other-window’. Optional argument *ARG* specifies ‘other’ window to speak.
- emacspeak-speak-page** (**&optional** *ARG*) [Interactive Command]  
*control e [*  
 Speak a page. With prefix *ARG*, speaks rest of current page. Negative prefix arg will read from start of current page to point. If option ‘voice-lock-mode’ is on, then it will use any defined personality.
- emacspeak-speak-page-interactively** () [Interactive Command]  
*control e ]*  
 Speak the start of, rest of, or the entire page. ‘s’ to speak the start. ‘r’ to speak the rest. any other key to speak entire page.
- emacspeak-speak-paragraph** (**&optional** *ARG*) [Interactive Command]  
*control e {*  
 Speak paragraph. With prefix arg, speaks rest of current paragraph. Negative prefix arg will read from start of current paragraph to point. If voice-lock-mode is on, then it will use any defined personality.
- emacspeak-speak-paragraph-interactively** () [Interactive Command]  
*control e cap P*  
 Speak the start of, rest of, or the entire paragraph. ‘s’ to speak the start. ‘r’ to speak the rest. any other key to speak entire paragraph.

**emacspeak-speak-predefined-window** (**&optional ARG**) [Interactive Command]  
*control e 0 control e 1 control e 2 control e 3 control e 4 control e 5 control e 6 control e 7 control e 8 control e 9*

Speak one of the first 10 windows on the screen. Speaks entire window irrespective of point. In general, you'll never have Emacs split the screen into more than two or three. Argument ARG determines the 'other' window to speak. Semantics of 'other' is the same as for the builtin Emacs command 'other-window'.

**emacspeak-speak-previous-field** () [Interactive Command]  
*control e <*

Move to previous field and speak it.

**emacspeak-speak-previous-personality-chunk** () [Interactive Command]  
 Moves to the front of previous chunk having current personality. Speak that chunk after moving.

**emacspeak-speak-previous-window** () [Interactive Command]  
*control e control p*

Speak the previous window.

**emacspeak-speak-rectangle** (*START END*) [Interactive Command]  
*control e cap R*

Speak a rectangle of text. Rectangle is delimited by point and mark. When call from a program, arguments specify the START and END of the rectangle.

**emacspeak-speak-region** (*START END*) [Interactive Command]  
*control e r*

Speak region. Argument START and END specify region to speak.

**emacspeak-speak-rest-of-buffer** () [Interactive Command]  
*control e n*

Speak remainder of the buffer starting at point

**emacspeak-speak-sentence** (**&optional ARG**) [Interactive Command]  
 Speak current sentence. With prefix ARG, speaks the rest of the sentence from point. Negative prefix arg speaks from start of sentence to point.

**emacspeak-speak-set-display-table** (**&optional PREFIX**) [Interactive Command]

Sets up buffer specific speech display table that controls how special characters are spoken. Interactive prefix argument causes setting to be global.

**emacspeak-speak-sexp** (**&optional ARG**) [Interactive Command]  
*control e '*

Speak current sexp. With prefix ARG, speaks the rest of the sexp from point. Negative prefix arg speaks from start of sexp to point. If option 'voice-lock-mode' is on, then uses the personality.

- `emacspeak-speak-sexp-interactively ()` [Interactive Command]  
*control e "*  
 Speak the start of, rest of, or the entire sexp. 's' to speak the start. 'r' to speak the rest. any other key to speak entire sexp.
- `emacspeak-speak-skim-buffer ()` [Interactive Command]  
 Skim the current buffer a paragraph at a time.
- `emacspeak-speak-skim-next-paragraph ()` [Interactive Command]  
 Skim next paragraph.
- `emacspeak-speak-skim-paragraph ()` [Interactive Command]  
 Skim paragraph. Skimming a paragraph results in the speech speeding up after the first clause. Speech is scaled by the value of `dtk-speak-skim-scale`
- `emacspeak-speak-spaces-at-point ()` [Interactive Command]  
*control e CONTROL meta @*  
 Speak the white space at point.
- `emacspeak-speak-spell-current-word ()` [Interactive Command]  
 Spell word at point.
- `emacspeak-speak-this-personality-chunk ()` [Interactive Command]  
 Speak chunk of text around point that has current personality.
- `emacspeak-speak-time (&optional WORLD)` [Interactive Command]  
*control e t*  
 Speak the time. Optional interactive prefix arg 'C-u' invokes world clock. Timezone is specified using minibuffer completion. Second interactive prefix sets clock to new timezone.
- `emacspeak-speak-version ()` [Interactive Command]  
*control e cap V*  
 Announce version information for running emacspeak.
- `emacspeak-speak-voice-annotate-paragraphs ()` [Interactive Command]  
 Locate paragraphs and voice annotate the first word. Here, paragraph is taken to mean a chunk of text preceded by a blank line. Useful to do this before you listen to an entire buffer.
- `emacspeak-speak-window-information ()` [Interactive Command]  
*control e control w*  
 Speaks information about current window.
- `emacspeak-speak-word (&optional ARG)` [Interactive Command]  
*control e w*  
 Speak current word. With prefix ARG, speaks the rest of the word from point. Negative prefix arg speaks from start of word to point. If executed on the same buffer position a second time, the word is spelt instead of being spoken.

- emacspeak-speak-word-interactively** () [Interactive Command]  
 Speak the start of, rest of, or the entire word. 's' to speak the start. 'r' to speak the rest. any other key to speak entire word.
- emacspeak-speak-world-clock** (*ZONE* &optional *SET*) [Interactive Command]  
 Display current date and time for specified zone. Optional second arg 'set' sets the TZ environment variable as well.
- emacspeak-switch-to-reference-buffer** () [Interactive Command]  
 Switch back to buffer that generated completions.
- emacspeak-toggle-action-mode** (&optional *PREFIX*) [Interactive Command]  
 Toggle state of Emacspeak action mode. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result.
- emacspeak-toggle-audio-indentation** (&optional *PREFIX*) [Interactive Command]  
*control e d i*  
 Toggle state of Emacspeak audio indentation. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result. Specifying the method of indentation as 'tones' results in the Dectalk producing a tone whose length is a function of the line's indentation. Specifying 'speak' results in the number of initial spaces being spoken.
- emacspeak-toggle-character-echo** (&optional *PREFIX*) [Interactive Command]  
*control e d k*  
 Toggle state of Emacspeak character echo. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result.
- emacspeak-toggle-comint-autospeak** (&optional *PREFIX*) [Interactive Command]  
*control e control q*  
 Toggle state of Emacspeak comint autospeak. When turned on, comint output is automatically spoken. Turn this on if you want your shell to speak its results. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result.
- emacspeak-toggle-line-echo** (&optional *PREFIX*) [Interactive Command]  
*control e d l*  
 Toggle state of Emacspeak line echo. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result.
- emacspeak-toggle-mail-alert** (&optional *PREFIX*) [Interactive Command]  
*control e meta m*  
 Toggle state of Emacspeak mail alert. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result. Turning on this option results in Emacspeak producing an auditory icon indicating the arrival of new mail when displaying the mode line.

**emacspeak-toggle-show-point** (**&optional** *PREFIX*) [Interactive Command]  
*control e control d*

Toggle state of Emacspeak-show-point. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result.

**emacspeak-toggle-speak-line-invert-filter** [Interactive Command]  
**(**&optional** *PREFIX*)**  
*control e \*

Toggle state of how column filter is interpreted. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result.

**emacspeak-toggle-speak-messages** (**&optional** *PREFIX*) [Interactive Command]  
*control e q*

Toggle the state of whether emacspeak echoes messages.

**emacspeak-toggle-which-function** (**&optional** *PREFIX*) [Interactive Command]  
*control e meta w*

Toggle state of Emacspeak which function mode. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result.

**emacspeak-toggle-word-echo** (**&optional** *PREFIX*) [Interactive Command]  
*control e d w*

Toggle state of Emacspeak word echo. Interactive *PREFIX* arg means toggle the global default value, and then set the current local value to the result.

**emacspeak-use-customized-blink-paren** () [Interactive Command]  
 A customized blink-paren to speak matching opening paren. We need to call this in case Emacs is anal and loads its own builtin blink-paren function which does not talk.

**emacspeak-voicify-rectangle** (*START END* **&optional** *PERSONALITY*) [Interactive Command]

Voicify the current rectangle. When calling from a program,arguments are *START END* personality Prompts for *PERSONALITY* with completion when called interactively.

**emacspeak-voicify-region** (*START END* **&optional** *PERSONALITY*) [Interactive Command]

Voicify the current region. When calling from a program,arguments are *START END* personality. Prompts for *PERSONALITY* with completion when called interactively.

**emacspeak-zap-tts** () [Interactive Command]  
*control e d z*

Send this command to the TTS directly.

## 11.50 emacspeak-speedbar

Commentary: This module advises speedbar.el for use with Emacs. The latest speedbar can be obtained from <ftp://ftp.ultranet.com/pub/zappo/> This module ensures that speedbar works smoothly outside a windowing system in addition to speech enabling all interactive commands. Emacspeak also adds an Emacspeak environment specific entry point to speedbar –emacspeak-speedbar-goto-speedbar– and binds this

Automatically generated documentation for commands defined in module emacspeak-speedbar.

**emacspeak-speedbar-click** () [Interactive Command]

Does the equivalent of the mouse click from the keyboard

**emacspeak-speedbar-goto-speedbar** () [Interactive Command]

Switch to the speedbar

## 11.51 emacspeak-table-ui

Automatically generated documentation for commands defined in module emacspeak-table-ui.

**emacspeak-table-copy-current-element-to-kill-** [Interactive Command]

**ring**

()

Copy current table element to kill ring.

**emacspeak-table-copy-current-element-to-register** [Interactive Command]

(*REGISTER*)

Copy current table element to specified register.

**emacspeak-table-copy-to-clipboard** () [Interactive Command]

Copy table in current buffer to the table clipboard. Current buffer must be in emacspeak-table mode.

**emacspeak-table-display-table-in-region** (*START* [Interactive Command]

*END*)

*control e TAB*

Recognize tabular data in current region and display it in table browsing mode in a a separate buffer. emacspeak table mode is designed to let you browse tabular data using all the power of the two-dimensional spatial layout while giving you sufficient contextual information. The tables subdirectory of the emacspeak distribution contains some sample tables –these are the CalTrain schedules. Execute command ‘describe-mode’ bound to C-h m in a buffer that is in emacspeak table mode to read the documentation on the table browser.

**emacspeak-table-find-csv-file** (*FILENAME*) [Interactive Command]

Process a csv (comma separated values) file. The processed data and presented using emacspeak table navigation.

- emacspeak-table-find-file** (*FILENAME*) [Interactive Command]  
*control e control t*  
 Open a file containing table data and display it in table mode. emacspeak table mode is designed to let you browse tabular data using all the power of the two-dimensional spatial layout while giving you sufficient contextual information. The etc/tables subdirectory of the emacspeak distribution contains some sample tables –these are the CalTrain schedules. Execute command ‘describe-mode’ bound to C-h m in a buffer that is in emacspeak table mode to read the documentation on the table browser.
- emacspeak-table-get-entry-with-headers** (*ROW COLUMN &optional ROW-HEAD-P COL-HEAD-P*) [Interactive Command]  
 Return both row and column header and table element
- emacspeak-table-goto** (*ROW COLUMN*) [Interactive Command]  
 Prompt for a table cell coordinates and jump to it.
- emacspeak-table-goto-bottom** () [Interactive Command]  
 Goes to the bottom of the current column.
- emacspeak-table-goto-left** () [Interactive Command]  
 Goes to the left of the current row.
- emacspeak-table-goto-right** () [Interactive Command]  
 Goes to the right of the current row.
- emacspeak-table-goto-top** () [Interactive Command]  
 Goes to the top of the current column.
- emacspeak-table-next-column** (**&optional** *COUNT*) [Interactive Command]  
 Move to the next column if possible
- emacspeak-table-next-row** (**&optional** *COUNT*) [Interactive Command]  
 Move to the next row if possible
- emacspeak-table-paste-from-clipboard** () [Interactive Command]  
 Paste the emacspeak table clipboard into the current buffer. Use the major mode of this buffer to decide what kind of table markup to use.
- emacspeak-table-previous-column** (**&optional** *COUNT*) [Interactive Command]  
 Move to the previous column if possible
- emacspeak-table-previous-row** (**&optional** *COUNT*) [Interactive Command]  
 Move to the previous row if possible
- emacspeak-table-search** (**&optional** *WHAT*) [Interactive Command]  
 Search the table for matching elements. Interactively prompts for row or column to search and pattern to look for. If there is a match, makes the matching cell current. When called from a program, ‘what’ can be either ‘row’ or ‘column’.

|                                                                                                                                                                                   |                       |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-table-search-column ()</code>                                                                                                                                     | [Interactive Command] |
| Search in current table column.                                                                                                                                                   |                       |
| <code>emacspeak-table-search-headers ()</code>                                                                                                                                    | [Interactive Command] |
| Search the table row or column headers. Interactively prompts for row or column to search and pattern to look for. If there is a match, makes the matching row or column current. |                       |
| <code>emacspeak-table-search-row ()</code>                                                                                                                                        | [Interactive Command] |
| Search in current table row.                                                                                                                                                      |                       |
| <code>emacspeak-table-select-automatic-speaking-method ()</code>                                                                                                                  | [Interactive Command] |
| Interactively select the kind of automatic speech to produce when browsing table elements                                                                                         |                       |
| <code>emacspeak-table-sort-on-current-column ()</code>                                                                                                                            | [Interactive Command] |
| Sort table on current column.                                                                                                                                                     |                       |
| <code>emacspeak-table-speak-both-headers-and-element ()</code>                                                                                                                    | [Interactive Command] |
| Speak both row and column header and table element                                                                                                                                |                       |
| <code>emacspeak-table-speak-column-filtered (&amp;optional PREFIX)</code>                                                                                                         | [Interactive Command] |
| Speaks a table column after applying a specified column filter. Optional prefix arg prompts for a new filter.                                                                     |                       |
| <code>emacspeak-table-speak-column-header-and-element ()</code>                                                                                                                   | [Interactive Command] |
| Speak column header and table element                                                                                                                                             |                       |
| <code>emacspeak-table-speak-coordinates ()</code>                                                                                                                                 | [Interactive Command] |
| Speak current table coordinates.                                                                                                                                                  |                       |
| <code>emacspeak-table-speak-current-element ()</code>                                                                                                                             | [Interactive Command] |
| Speak current table element                                                                                                                                                       |                       |
| <code>emacspeak-table-speak-dimensions ()</code>                                                                                                                                  | [Interactive Command] |
| Speak current table dimensions.                                                                                                                                                   |                       |
| <code>emacspeak-table-speak-row-filtered (&amp;optional PREFIX)</code>                                                                                                            | [Interactive Command] |
| Speaks a table row after applying a specified row filter. Optional prefix arg prompts for a new filter.                                                                           |                       |
| <code>emacspeak-table-speak-row-header-and-element ()</code>                                                                                                                      | [Interactive Command] |
| Speak row header and table element                                                                                                                                                |                       |
| <code>emacspeak-table-ui-filter-load (FILE)</code>                                                                                                                                | [Interactive Command] |
| Load saved filter settings.                                                                                                                                                       |                       |

**emacspeak-table-ui-filter-save** (*FILE*) [Interactive Command]  
 Save out filter settings.

**emacspeak-table-view-csv-buffer** (&optional *BUFFER-NAME*) [Interactive Command]  
 Process a csv (comma separated values) data. The processed data and presented using emacspeak table navigation.

## 11.52 emacspeak-tabulate

Automatically generated documentation for commands defined in module emacspeak-tabulate.

**emacspeak-tabulate-region** (*START END* &optional *MARK-FIELDS*) [Interactive Command]  
*control e i*  
 Voicifies the white-space of a table if one found. Optional interactive prefix arg mark-fields specifies if the header row information is used to mark fields in the white-space.

## 11.53 emacspeak-tapestry

Automatically generated documentation for commands defined in module emacspeak-tapestry.

**emacspeak-tapestry-describe-tapestry** (&optional *DETAILS*) [Interactive Command]  
*control e meta t*  
 Describe the current layout of visible buffers in current frame. Use interactive prefix arg to get coordinate positions of the displayed buffers.

**emacspeak-tapestry-select-window-by-name** (*BUFFER-NAME*) [Interactive Command]  
*control e cap W*  
 Select window by the name of the buffer it displays. This is useful when using modes like ECB or the new GDB UI where you want to preserve the window layout but quickly switch to a window by name.

## 11.54 emacspeak-tar

Automatically generated documentation for commands defined in module emacspeak-tar.

**emacspeak-tar-speak-file-date** () [Interactive Command]  
 Speak date of file current entry

**emacspeak-tar-speak-file-permissions** () [Interactive Command]  
 Speak permissions of file current entry

**emacspeak-tar-speak-file-size** () [Interactive Command]  
 Speak size of file current entry

## 11.55 emacspeak-tetris

Automatically generated documentation for commands defined in module emacspeak-tetris.

|                                                                      |                       |
|----------------------------------------------------------------------|-----------------------|
| <code>emacspeak-tetris-goto-bottom-row ()</code>                     | [Interactive Command] |
| Move to and speak bottom row                                         |                       |
| <code>emacspeak-tetris-goto-top-row ()</code>                        | [Interactive Command] |
| Move to and speak the top row                                        |                       |
| <code>emacspeak-tetris-speak-column (&amp;optional X)</code>         | [Interactive Command] |
| Speak column –default is to speak current column                     |                       |
| <code>emacspeak-tetris-speak-coordinates ()</code>                   | [Interactive Command] |
| Speak current position                                               |                       |
| <code>emacspeak-tetris-speak-current-shape ()</code>                 | [Interactive Command] |
| Speak current shape                                                  |                       |
| <code>emacspeak-tetris-speak-current-shape-and-coordinates ()</code> | [Interactive Command] |
| Speak shape orientation and coordinates                              |                       |
| <code>emacspeak-tetris-speak-next-shape ()</code>                    | [Interactive Command] |
| Speak next shape                                                     |                       |
| <code>emacspeak-tetris-speak-row ()</code>                           | [Interactive Command] |
| Speak current tetris row                                             |                       |
| <code>emacspeak-tetris-speak-row-number ()</code>                    | [Interactive Command] |
| Speak where on the tetris board we are                               |                       |
| <code>emacspeak-tetris-speak-score ()</code>                         | [Interactive Command] |
| Speak the score                                                      |                       |
| <code>emacspeak-tetris-speak-x-coordinate ()</code>                  | [Interactive Command] |
| Speak current position                                               |                       |

## 11.56 emacspeak-tnt

Commentary: Speech-enables TNT – the Emacs AOL Instant Messenger client

Automatically generated documentation for commands defined in module emacspeak-tnt.

|                                                                    |                       |
|--------------------------------------------------------------------|-----------------------|
| <code>emacspeak-tnt-toggle-autospeak (&amp;optional PREFIX)</code> | [Interactive Command] |
| Toggle TNT autospeak for this chat session.                        |                       |

## 11.57 emacspeak-url-template

Commentary: It is often useful to have “parameterized hot list entries” i.e., hotlist entries that are “templates” for the actual URL. The user provides values for the parameterized portions of the URL e.g. the date. See Section 11.76 [URL Templates], page 126, for details on the URL templates that are presently defined.

Automatically generated documentation for commands defined in module emacspeak-url-template.

**emacspeak-url-template-fetch** (**&optional** *DOCUMENTATION*) [Interactive Command]

*control e u*

Fetch a pre-defined resource. Use Emacs completion to obtain a list of available resources. Resources typically prompt for the relevant information before completing the request. Optional interactive prefix arg displays documentation for specified resource.

**emacspeak-url-template-help** () [Interactive Command]

Display documentation for a URL template. Use Emacs completion to obtain a list of available resources.

**emacspeak-url-template-load** (*FILE*) [Interactive Command]

Load URL template resources from specified location.

**emacspeak-url-template-save** (*FILE*) [Interactive Command]

Save out url templates.

## 11.58 emacspeak-view-process

Automatically generated documentation for commands defined in module emacspeak-view-process.

**emacspeak-view-process-goto-current-field-next-line** () [Interactive Command]

Set point to the current field in the next line.

**emacspeak-view-process-speak-current-field** () [Interactive Command]

Speak current field

## 11.59 emacspeak-view

Automatically generated documentation for commands defined in module emacspeak-view.

**emacspeak-view-line-to-top** () [Interactive Command]

Moves current line to top of window

## 11.60 emacspeak-vm

Automatically generated documentation for commands defined in module emacspeak-vm.

- `emacspeak-vm-browse-message` () [Interactive Command]  
Browse an email message –read it paragraph at a time.
- `emacspeak-vm-catch-up-all-messages` () [Interactive Command]  
Mark all messages in folder to be deleted. Use with caution.
- `emacspeak-vm-locate-subject-line` () [Interactive Command]  
Locates the subject line in a message being read. Useful when you’re reading a message that has been forwarded multiple times.
- `emacspeak-vm-mode-line` () [Interactive Command]  
VM mode line information.
- `emacspeak-vm-next-button` (*N*) [Interactive Command]  
Move point to *N* buttons forward. If *N* is negative, move backward instead.
- `emacspeak-vm-speak-labels` () [Interactive Command]  
Speak a message’s labels
- `emacspeak-vm-speak-message` () [Interactive Command]  
Move point to the message body.
- `emacspeak-vm-yank-header` () [Interactive Command]  
Yank specified header into kill ring.

## 11.61 emacspeak-w3

Commentary: Ensure that speech support for W3 gets installed and loaded correctly. The emacs W3 browser comes with builtin support for Emacspeak and ACSS

Automatically generated documentation for commands defined in module emacspeak-w3.

- `emacspeak-realaudio` (**&optional** *IGNORED*) [Interactive Command]  
*control e* :

Start or control streaming audio including MP3 and realaudio. If using ‘TRPlayer’ as the player, accepts trplayer control commands if a stream is already playing. Otherwise, the playing stream is simply stopped. If no stream is playing, this command prompts for a realaudio resource. Realaudio resources can be specified either as a Realaudio URL, the location of a local Realaudio file, or as the name of a local Realaudio metafile. Realaudio resources you have played in this session are available in the minibuffer history. The default is to play the resource you played most recently. Emacspeak uses the contents of the directory specified by variable `emacspeak-realaudio-shortcuts-directory` to offer a set of completions. Hit space to use this completion list.

If using TRPlayer, you can either give one-shot commands using command `emacspeak-realaudio` available from anywhere on the audio desktop as ‘C-e :’.

Alternatively, switch to buffer *\*realaudio\** using ‘C-e ;;’ if you wish to issue many navigation commands. Note that buffer *\*realaudio\** uses a special major mode that provides the various navigation commands via single keystrokes.

- `emacspeak-w3-curl-url-under-point ()` [Interactive Command]  
 Display contents of URL under point using Curl and W3. The document is displayed in a separate buffer.
- `emacspeak-w3-customize-base-uri-pronunciation ()` [Interactive Command]  
 Defines custom buffer local pronunciation for base URL.
- `emacspeak-w3-do-onclick ()` [Interactive Command]  
 Do onclick action.
- `emacspeak-w3-javascript-follow-link ()` [Interactive Command]  
 Follow URL hidden inside a javascript link
- `emacspeak-w3-jump-to-submit ()` [Interactive Command]  
 Jump to next available submit button.
- `emacspeak-w3-lynx-url-under-point ()` [Interactive Command]  
 Display contents of URL under point using LYNX. The document is displayed in a separate buffer. Note that the hyperlinks in that display are not active— this facility is present only to help me iron out the remaining problems with the table structure extraction code in W3.
- `emacspeak-w3-next-block ()` [Interactive Command]  
 Move by block level displays.
- `emacspeak-w3-next-doc-element (&optional COUNT)` [Interactive Command]  
 Move forward to the next document element. Optional interactive prefix argument COUNT specifies by how many elements to move.
- `emacspeak-w3-next-parsed-item ()` [Interactive Command]  
 Move to and speak next parsed item.
- `emacspeak-w3-previous-doc-element (&optional COUNT)` [Interactive Command]  
 Move back to the previous document element. Optional interactive prefix argument COUNT specifies by how many elements to move.
- `emacspeak-w3-show-anchor-class ()` [Interactive Command]  
 Display any class attributes set on corresponding anchor element.
- `emacspeak-w3-show-http-headers ()` [Interactive Command]  
 Show HTTP headers using lwp-request
- `emacspeak-w3-speak-next-block ()` [Interactive Command]  
 Move to next block and speak it.
- `emacspeak-w3-speak-next-element ()` [Interactive Command]  
 Speak next document element.

**emacspeak-w3-speak-this-element** () [Interactive Command]  
 Speak document element under point.

**emacspeak-w3-toggle-table-borders** () [Interactive Command]  
 Toggle drawing of W3 table borders

## 11.62 emacspeak-w3m

Commentary:

Automatically generated documentation for commands defined in module emacspeak-w3m.

**emacspeak-m-player** (*RESOURCE* &**optional** *PLAY-LIST* [Interactive Command]  
*NOSELECT*)  
 Play specified resource using m-player. Optional prefix argument play-list interprets resource as a play-list. Resource is a media resource or playlist containing media resources. The player is placed in a buffer in emacspeak-m-player-mode.

**emacspeak-w3m-preview-this-buffer** () [Interactive Command]  
 Preview this buffer in w3m.

## 11.63 emacspeak-we

Commentary: we is for webedit Invoke XSLT to edit/transform Web pages before they get rendered. we makes emacspeak's webedit layer independent of a given Emacs web browser like W3 or W3M This module will use the abstraction provided by browse-url to handle Web pages. Module emacspeak-weutils provides the needed additional abstractions not already covered by browse-url

Automatically generated documentation for commands defined in module emacspeak-we.

**emacspeak-we-class-filter-and-follow** (*CLASS URL*) [Interactive Command]  
 Follow url and point, and filter the result by specified class. Class can be set locally for a buffer, and overridden with an interactive prefix arg. If there is a known rewrite url rule, that is used as well.

**emacspeak-we-count-matches** (*URL LOCATOR*) [Interactive Command]  
 Count matches for locator in Web page.

**emacspeak-we-count-nested-tables** (*URL*) [Interactive Command]  
 Count nested tables in Web page.

**emacspeak-we-count-tables** (*URL*) [Interactive Command]  
 Count tables in Web page.

**emacspeak-we-extract-by-class** (*CLASS URL* [Interactive Command]  
 &**optional** *SPEAK*)  
 Extract elements having specified class attribute from HTML. Extracts specified elements from current WWW page and displays it in a separate buffer. Interactive use provides list of class values as completion.

**emacspeak-we-extract-by-class-list** (*CLASSES URL* [Interactive Command]  
**&optional SPEAK**)

Extract elements having class specified in list ‘classes’ from HTML. Extracts specified elements from current WWW page and displays it in a separate buffer. Interactive use provides list of class values as completion.

**emacspeak-we-extract-by-id** (*ID URL &optional* [Interactive Command]  
**SPEAK**)

Extract elements having specified id attribute from HTML. Extracts specified elements from current WWW page and displays it in a separate buffer. Interactive use provides list of id values as completion.

**emacspeak-we-extract-by-id-list** (*IDS URL* [Interactive Command]  
**&optional SPEAK**)

Extract elements having id specified in list ‘ids’ from HTML. Extracts specified elements from current WWW page and displays it in a separate buffer. Interactive use provides list of id values as completion.

**emacspeak-we-extract-matching-urls** (*PATTERN URL* [Interactive Command]  
**&optional SPEAK**)

Extracts links whose URL matches pattern.

**emacspeak-we-extract-media-streams** (*URL &optional* [Interactive Command]  
**SPEAK**)

Extract links to media streams. operate on current web page when in a browser buffer; otherwise prompt for url. Optional arg ‘speak’ specifies if the result should be spoken automatically.

**emacspeak-we-extract-media-streams-under-point** () [Interactive Command]  
 In browser buffers, extract media streams from url under point.

**emacspeak-we-extract-nested-table** (*INDEX URL* [Interactive Command]  
**&optional SPEAK**)

Extract nested table specified by ‘table-index’. Default is to operate on current web page when in a browser buffer; otherwise prompt for URL. Optional arg ‘speak’ specifies if the result should be spoken automatically.

**emacspeak-we-extract-nested-table-list** (*TABLES* [Interactive Command]  
*URL &optional SPEAK*)

Extract specified list of tables from a Web page.

**emacspeak-we-extract-print-streams** (*URL &optional* [Interactive Command]  
**SPEAK**)

Extract links to printable streams. operate on current web page when in a browser buffer; otherwise prompt for url. Optional arg ‘speak’ specifies if the result should be spoken automatically.

**emacspeak-we-extract-table-by-match** (*MATCH URL* [Interactive Command]  
**&optional SPEAK**)

Extract table containing specified match. Optional arg url specifies the page to extract content from.

**emacspeak-we-extract-table-by-position** (*POSITION* [Interactive Command]  
*URL &optional SPEAK*)

Extract table at specified position. Default is to extract from current page.

**emacspeak-we-extract-tables-by-match-list** [Interactive Command]  
*(MATCH-LIST URL &optional SPEAK)*

Extract specified tables from a WWW page. Tables are specified by containing match pattern found in the match list.

**emacspeak-we-extract-tables-by-position-list** [Interactive Command]  
*(POSITIONS URL &optional SPEAK)*

Extract specified list of nested tables from a WWW page. Tables are specified by their position in the list of nested tables found in the page.

**emacspeak-we-follow-and-filter-by-id** (*ID*) [Interactive Command]

Follow url and point, and filter the result by specified id. Id can be set locally for a buffer, and overridden with an interactive prefix arg. If there is a known rewrite url rule, that is used as well.

**emacspeak-we-junk-by-class-list** (*CLASSES URL* [Interactive Command]  
*&optional SPEAK*)

Junk elements having class specified in list ‘classes’ from HTML. Extracts specified elements from current WWW page and displays it in a separate buffer. Interactive use provides list of class values as completion.

**emacspeak-we-style-filter** (*STYLE URL &optional* [Interactive Command]  
*SPEAK*)

Extract elements matching specified style from HTML. Extracts specified elements from current WWW page and displays it in a separate buffer. Optional arg url specifies the page to extract contents from.

**emacspeak-we-toggle-xsl-keep-result** () [Interactive Command]

Toggle xsl keep result flag.

**emacspeak-we-url-expand-and-execute** () [Interactive Command]

Applies buffer-specific URL expander/executor function.

**emacspeak-we-url-rewrite-and-follow** (*&optional* [Interactive Command]  
*PROMPT*)

Apply a url rewrite rule as specified in the current buffer before following link under point. If no rewrite rule is defined, first prompt for one. Rewrite rules are of the form ‘(from to)’ where from and to are strings. Typically, the rewrite rule is automatically set up by Emacspeak tools like websearch where a rewrite rule is known. Rewrite rules are useful in jumping directly to the printer friendly version of an article for example. Optional interactive prefix arg prompts for a rewrite rule even if one is already defined.

- emacspeak-we-xpath-filter-and-follow** (**&optional** *PROMPT*) [Interactive Command]  
 Follow url and point, and filter the result by specified xpath. XPath can be set locally for a buffer, and overridden with an interactive prefix arg. If there is a known rewrite url rule, that is used as well.
- emacspeak-we-xpath-junk-and-follow** (**&optional** *PROMPT*) [Interactive Command]  
 Follow url and point, and filter the result by junking elements specified by xpath. XPath can be set locally for a buffer, and overridden with an interactive prefix arg. If there is a known rewrite url rule, that is used as well.
- emacspeak-we-xsl-toggle** () [Interactive Command]  
 Toggle application of XSL transformations.
- emacspeak-we-xslt-apply** (*XSL*) [Interactive Command]  
 Apply specified transformation to current Web page.
- emacspeak-we-xslt-filter** (*PATH URL &optional SPEAK*) [Interactive Command]  
 Extract elements matching specified XPath path locator from Web page – default is the current page being viewed.
- emacspeak-we-xslt-junk** (*PATH URL &optional SPEAK*) [Interactive Command]  
 Junk elements matching specified locator.
- emacspeak-we-xslt-select** (*XSL*) [Interactive Command]  
 Select XSL transformation applied to Web pages before they are displayed .

## 11.64 emacspeak-websearch

Commentary: This module provides utility functions for searching the WWW

Automatically generated documentation for commands defined in module emacspeak-websearch.

- emacspeak-websearch-accessible-google** (*QUERY*) [Interactive Command]  
 Google Accessible Search – see <http://labs.google.com/accessible>
- emacspeak-websearch-alltheweb-search** (*QUERY*) [Interactive Command]  
 Perform an AllTheWeb search.
- emacspeak-websearch-altavista-search** (*QUERY*) [Interactive Command]  
 Perform an Altavista search
- emacspeak-websearch-amazon-search** () [Interactive Command]  
 Amazon search.
- emacspeak-websearch-ask-jeeves** (*QUERY*) [Interactive Command]  
 Ask Jeeves for the answer.

|                                                                                                                                                                                                                                                                                            |                       |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-websearch-bbc-search</code> ( <i>QUERY</i> )                                                                                                                                                                                                                               | [Interactive Command] |
| Search BBC archives.                                                                                                                                                                                                                                                                       |                       |
| <code>emacspeak-websearch-biblio-search</code> ( <i>QUERY</i> )                                                                                                                                                                                                                            | [Interactive Command] |
| Search Computer Science Bibliographies.                                                                                                                                                                                                                                                    |                       |
| <code>emacspeak-websearch-blinkx-search</code> ( <i>QUERY</i> )                                                                                                                                                                                                                            | [Interactive Command] |
| BlinkX RSS Generator.                                                                                                                                                                                                                                                                      |                       |
| <code>emacspeak-websearch-britannica-search</code> ( <i>QUERY</i> )                                                                                                                                                                                                                        | [Interactive Command] |
| Search Encyclopedia Britannica.                                                                                                                                                                                                                                                            |                       |
| <code>emacspeak-websearch-citeseer-search</code> ( <i>TERM</i> )                                                                                                                                                                                                                           | [Interactive Command] |
| Perform a CiteSeer search.                                                                                                                                                                                                                                                                 |                       |
| <code>emacspeak-websearch-company-news</code> ( <i>TICKER</i><br>& <b>optional PREFIX</b> )                                                                                                                                                                                                | [Interactive Command] |
| Perform an company news lookup. Retrieves company news, research, profile, insider trades, or upgrades/downgrades.                                                                                                                                                                         |                       |
| <code>emacspeak-websearch-cpan-search</code> ( <i>QUERY</i> )                                                                                                                                                                                                                              | [Interactive Command] |
| Search CPAN Comprehensive Perl Archive Network Site.                                                                                                                                                                                                                                       |                       |
| <code>emacspeak-websearch-ctan-search</code> ( <i>QUERY</i> )                                                                                                                                                                                                                              | [Interactive Command] |
| Search CTAN Comprehensive TeX Archive Network Site.                                                                                                                                                                                                                                        |                       |
| <code>emacspeak-websearch-dictionary-hypertext-webster-search</code> ( <i>QUERY</i> )                                                                                                                                                                                                      | [Interactive Command] |
| Search the Webster Dictionary.                                                                                                                                                                                                                                                             |                       |
| <code>emacspeak-websearch-dispatch</code> (& <b>optional PREFIX</b><br><i>control e ?</i> )                                                                                                                                                                                                | [Interactive Command] |
| Launches specific websearch queries. Press '?' to list available search engines. Once selected, the selected searcher prompts for additional information as appropriate. When using supported browsers, this interface attempts to speak the most relevant information on the result page. |                       |
| <code>emacspeak-websearch-display-form</code> ( <i>FORM-MARKUP</i> )                                                                                                                                                                                                                       | [Interactive Command] |
| Display form specified by form-markup.                                                                                                                                                                                                                                                     |                       |
| <code>emacspeak-websearch-ebay-search</code> ()                                                                                                                                                                                                                                            | [Interactive Command] |
| Ebay search.                                                                                                                                                                                                                                                                               |                       |
| <code>emacspeak-websearch-emaps-search</code> ( <i>QUERY</i><br>& <b>optional USE-NEAR</b> )                                                                                                                                                                                               | [Interactive Command] |
| Perform EmapSpeak search. Query is a Google Maps query in plain English. Interactive prefix arg 'use-near' searches near our previously cached location.                                                                                                                                   |                       |
| <code>emacspeak-websearch-emapspeak-near-my-location</code> ( <i>QUERY</i> )                                                                                                                                                                                                               | [Interactive Command] |
| Perform search relative to 'my-location'.                                                                                                                                                                                                                                                  |                       |

|                                                                                                                                                                                                            |                       |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-websearch-exchange-rate-convertor</code><br>( <i>CONVERSION-SPEC</i> )                                                                                                                     | [Interactive Command] |
| Currency convertor.                                                                                                                                                                                        |                       |
| <code>emacspeak-websearch-foldoc-search</code> ( <i>QUERY</i> )                                                                                                                                            | [Interactive Command] |
| Perform a FolDoc search.                                                                                                                                                                                   |                       |
| <code>emacspeak-websearch-freshmeat-search</code> ( <i>QUERY</i> )                                                                                                                                         | [Interactive Command] |
| Search Freshmeat Site.                                                                                                                                                                                     |                       |
| <code>emacspeak-websearch-froogle</code> ( <i>QUERY &amp;optional</i><br><i>LOCAL-FLAG</i> )                                                                                                               | [Interactive Command] |
| Perform a Froogle search. Optional interactive prefix arg <i>local-flag</i> prompts for local area in which to search.                                                                                     |                       |
| <code>emacspeak-websearch-google</code> ( <i>QUERY &amp;optional</i><br><i>LUCKY</i> )                                                                                                                     | [Interactive Command] |
| Perform a Google search. Optional interactive prefix arg ‘lucky’ is equivalent to hitting the I’m Feeling Lucky button on Google.                                                                          |                       |
| <code>emacspeak-websearch-google-advanced</code> ()                                                                                                                                                        | [Interactive Command] |
| Present Google advanced search form simplified for speech interaction.                                                                                                                                     |                       |
| <code>emacspeak-websearch-google-feeling-lucky</code><br>( <i>QUERY</i> )                                                                                                                                  | [Interactive Command] |
| Do a I’m Feeling Lucky Google search.                                                                                                                                                                      |                       |
| <code>emacspeak-websearch-google-mobile</code> ( <i>QUERY</i> )                                                                                                                                            | [Interactive Command] |
| Google mobile search.                                                                                                                                                                                      |                       |
| <code>emacspeak-websearch-google-news</code> ()                                                                                                                                                            | [Interactive Command] |
| Invoke Google News url template.                                                                                                                                                                           |                       |
| <code>emacspeak-websearch-google-sak</code> ( <i>ENGINE QUERY</i> )                                                                                                                                        | [Interactive Command] |
| Perform a Google query against a specific index.                                                                                                                                                           |                       |
| <code>emacspeak-websearch-google-search-in-date-range</code><br>( )                                                                                                                                        | [Interactive Command] |
| Use this from inside the calendar to do Google date-range searches.                                                                                                                                        |                       |
| <code>emacspeak-websearch-google-specialize</code><br>( <i>SPECIALIZE QUERY</i> )                                                                                                                          | [Interactive Command] |
| Perform a specialized Google search. See the Google site for what is possible here:<br><a href="http://www.google.com/options/specialsearches.html">http://www.google.com/options/specialsearches.html</a> |                       |
| <code>emacspeak-websearch-google-usenet-advanced</code> ()                                                                                                                                                 | [Interactive Command] |
| Present Google Usenet advanced search form simplified for speech interaction.                                                                                                                              |                       |
| <code>emacspeak-websearch-gutenberg</code> ( <i>TYPE QUERY</i> )                                                                                                                                           | [Interactive Command] |
| Perform an Gutenberg search                                                                                                                                                                                |                       |

|                                                                                                                                                                                                                                                                                                                                                                                                                   |                       |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-websearch-help ()</code>                                                                                                                                                                                                                                                                                                                                                                          | [Interactive Command] |
| Displays key mapping used by Emacspeak Websearch.                                                                                                                                                                                                                                                                                                                                                                 |                       |
| <code>emacspeak-websearch-koders-search (QUERY)</code>                                                                                                                                                                                                                                                                                                                                                            | [Interactive Command] |
| Perform a koders.com search                                                                                                                                                                                                                                                                                                                                                                                       |                       |
| <code>emacspeak-websearch-map-yahoo-directions-search (QUERY &amp;optional MAP)</code>                                                                                                                                                                                                                                                                                                                            | [Interactive Command] |
| Get driving directions from Yahoo. With optional interactive prefix arg MAP shows the location map instead.                                                                                                                                                                                                                                                                                                       |                       |
| <code>emacspeak-websearch-merriam-webster-search (QUERY)</code>                                                                                                                                                                                                                                                                                                                                                   | [Interactive Command] |
| Search the Merriam Webster Dictionary.                                                                                                                                                                                                                                                                                                                                                                            |                       |
| <code>emacspeak-websearch-my-rss-search (QUERY)</code>                                                                                                                                                                                                                                                                                                                                                            | [Interactive Command] |
| My RSS search.                                                                                                                                                                                                                                                                                                                                                                                                    |                       |
| <code>emacspeak-websearch-news-yahoo (QUERY &amp;optional NO-RSS)</code>                                                                                                                                                                                                                                                                                                                                          | [Interactive Command] |
| Perform an Yahoo News search. Optional prefix arg no-rss scrapes information from HTML.                                                                                                                                                                                                                                                                                                                           |                       |
| <code>emacspeak-websearch-open-directory-search (QUERY)</code>                                                                                                                                                                                                                                                                                                                                                    | [Interactive Command] |
| Perform an Open Directory search                                                                                                                                                                                                                                                                                                                                                                                  |                       |
| <code>emacspeak-websearch-people-yahoo ()</code>                                                                                                                                                                                                                                                                                                                                                                  | [Interactive Command] |
| Perform an Yahoo people search                                                                                                                                                                                                                                                                                                                                                                                    |                       |
| <code>emacspeak-websearch-podscope ()</code>                                                                                                                                                                                                                                                                                                                                                                      | [Interactive Command] |
| Perform a PodScope search to locate podcasts.                                                                                                                                                                                                                                                                                                                                                                     |                       |
| <code>emacspeak-websearch-podzinger-search (QUERY)</code>                                                                                                                                                                                                                                                                                                                                                         | [Interactive Command] |
| Podzinger RSS Generator.                                                                                                                                                                                                                                                                                                                                                                                          |                       |
| <code>emacspeak-websearch-quotes-yahoo-search (QUERY &amp;optional PREFIX)</code>                                                                                                                                                                                                                                                                                                                                 | [Interactive Command] |
| Perform a Quotes Yahoo . Default tickers to look up is taken from variable <code>emacspeak-websearch-personal-portfolio</code> . Default is to present the data in emacspeak's table browsing mode –optional interactive prefix arg causes data to be displayed as a Web page. You can customize the defaults by setting variable <code>emacspeak-websearch-quotes-yahoo-options</code> to an appropriate string. |                       |
| <code>emacspeak-websearch-recorded-books-search ()</code>                                                                                                                                                                                                                                                                                                                                                         | [Interactive Command] |
| Present advanced search form for recorded books.                                                                                                                                                                                                                                                                                                                                                                  |                       |
| <code>emacspeak-websearch-rpm-find (QUERY)</code>                                                                                                                                                                                                                                                                                                                                                                 | [Interactive Command] |
| Search RPM catalog site.                                                                                                                                                                                                                                                                                                                                                                                          |                       |
| <code>emacspeak-websearch-shoutcast-search ()</code>                                                                                                                                                                                                                                                                                                                                                              | [Interactive Command] |
| Shoutcast search.                                                                                                                                                                                                                                                                                                                                                                                                 |                       |

|                                                                                                               |                       |
|---------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-websearch-software-search</code> ()                                                           | [Interactive Command] |
| Search SourceForge, Freshmeat and other sites.                                                                |                       |
| <code>emacspeak-websearch-sourceforge-search</code> ( <i>QUERY</i> )                                          | [Interactive Command] |
| Search SourceForge Site.                                                                                      |                       |
| <code>emacspeak-websearch-swik-search</code> ( <i>QUERY</i> )                                                 | [Interactive Command] |
| Search swik software community site.                                                                          |                       |
| <code>emacspeak-websearch-technorati</code> ( <i>QUERY</i> )                                                  | [Interactive Command] |
| Perform a Technorati tag search.                                                                              |                       |
| <code>emacspeak-websearch-teoma</code> ( <i>QUERY</i> )                                                       | [Interactive Command] |
| Perform an Teoma search.                                                                                      |                       |
| <code>emacspeak-websearch-usenet</code> ( <i>GROUP &amp;optional PREFIX</i> )                                 | [Interactive Command] |
| <i>control e cap U</i>                                                                                        |                       |
| Prompt and browse a Usenet newsgroup. Optional interactive prefix arg results in prompting for a search term. |                       |
| <code>emacspeak-websearch-usenet-search</code> ( <i>GROUP</i> )                                               | [Interactive Command] |
| Search a Usenet newsgroup.                                                                                    |                       |
| <code>emacspeak-websearch-w3c-search</code> ( <i>QUERY</i> )                                                  | [Interactive Command] |
| Search the W3C Site.                                                                                          |                       |
| <code>emacspeak-websearch-weather</code> ( <i>QUERY</i> )                                                     | [Interactive Command] |
| Get weather forecast for specified zip code.                                                                  |                       |
| <code>emacspeak-websearch-wikipedia-search</code> ( <i>QUERY</i> )                                            | [Interactive Command] |
| Search Wikipedia                                                                                              |                       |
| <code>emacspeak-websearch-yahoo</code> ( <i>QUERY</i> )                                                       | [Interactive Command] |
| Perform an Yahoo search                                                                                       |                       |
| <code>emacspeak-websearch-yahoo-historical-chart</code> ( <i>TICKER &amp;optional AS-HTML</i> )               | [Interactive Command] |
| Look up historical stock data. Optional second arg as-html processes the results as HTML rather than data.    |                       |
| <code>emacspeak-websearch-youtube</code> ( <i>QUERY</i> )                                                     | [Interactive Command] |
| Search YouTube                                                                                                |                       |

## 11.65 emacspeak-webutils

Commentary: This module provides common Web utilities for emacspeak. This is to avoid duplication of code between emacspeak-w3.el and emacspeak-w3m.el

Automatically generated documentation for commands defined in module emacspeak-webutils.

|                                                                                                                                                                                                                        |                       |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-atom-browse</code> ( <i>FEED</i> )<br>Browse specified ATOM feed.                                                                                                                                      | [Interactive Command] |
| <code>emacspeak-opml-display</code> ( <i>OPML-URL</i> & <b>optional</b><br><i>SPEAK</i> )<br>Retrieve and display OPML URL.                                                                                            | [Interactive Command] |
| <code>emacspeak-rss-browse</code> ( <i>FEED</i> )<br><i>control e control u</i><br>Browse specified RSS feed.                                                                                                          | [Interactive Command] |
| <code>emacspeak-webutils-atom-display</code> ( <i>FEED-URL</i> )<br>Display ATOM feed.                                                                                                                                 | [Interactive Command] |
| <code>emacspeak-webutils-fv</code> ( <i>FEED-URL</i> )<br>Display RSS or ATOM feed.                                                                                                                                    | [Interactive Command] |
| <code>emacspeak-webutils-google-extract-from-cache</code><br>( <b>&amp;optional</b> <i>PREFIX</i> )<br>Extract current page from the Google cache. With a prefix argument, extracts url under point.                   | [Interactive Command] |
| <code>emacspeak-webutils-google-on-this-site</code> ()<br>Perform a google search restricted to the current WWW site.                                                                                                  | [Interactive Command] |
| <code>emacspeak-webutils-google-similar-to-this-page</code><br>( <i>URL</i> )<br>Ask Google to find documents similar to this one.                                                                                     | [Interactive Command] |
| <code>emacspeak-webutils-google-who-links-to-this-page</code><br>()<br>Perform a google search to locate documents that link to the current page.                                                                      | [Interactive Command] |
| <code>emacspeak-webutils-jump-to-title-in-content</code> ()<br>Jumps to the title in web document. The first time it is called, it jumps to the first instance of the title. Repeated calls jump to further instances. | [Interactive Command] |
| <code>emacspeak-webutils-open-in-other-browser</code> ()<br>Opens link in alternate browser. If using default browser is w3, uses w3m and vice-versa                                                                   | [Interactive Command] |
| <code>emacspeak-webutils-play-media-at-point</code> ()<br>Play media url under point                                                                                                                                   | [Interactive Command] |

- `emacspeak-webutils-rss-display` (*FEED-URL*) [Interactive Command]  
 Display RSS feed.
- `emacspeak-webutils-transcode-current-url-via-google` (**&optional** *UNTRANSCODE*) [Interactive Command]  
 Transcode current URL via Google. Reverse effect with prefix arg.
- `emacspeak-webutils-transcode-via-google` (**&optional** *UNTRANSCODE*) [Interactive Command]  
 Transcode URL under point via Google. Reverse effect with prefix arg for links on a transcoded page.
- `emacspeak-webutils-view-feed-via-google-reader` () [Interactive Command]  
 Pulls feed under point via Google Reader.

### 11.66 emacspeak-widget

Commentary: This module implements the necessary extensions to provide talking widgets.

Automatically generated documentation for commands defined in module emacspeak-widget.

- `emacspeak-widget-browse-widget-interactively` () [Interactive Command]  
 Allows you to browse a widget
- `emacspeak-widget-help` () [Interactive Command]  
 Speak help for widget under point.
- `emacspeak-widget-summarize-parent` () [Interactive Command]  
 Summarize parent of widget at point.
- `emacspeak-widget-summarize-widget-under-point` (**&optional** *LEVEL*) [Interactive Command]  
 Summarize a widget if any under point. Optional interactive prefix specifies how many levels to go up from current widget before summarizing.
- `emacspeak-widget-update-from-minibuffer` (*POINT*) [Interactive Command]  
 Sets widget at point by invoking its prompter.

### 11.67 emacspeak-wizards

Commentary: Contains various wizards for the Emacspeak desktop.

Automatically generated documentation for commands defined in module emacspeak-wizards.

- `emacspeak-annotate-add-annotation` (**&optional** *RESET*) [Interactive Command]  
 Add annotation to the annotation working buffer. Prompt for annotation buffer if not already set. Interactive prefix arg ‘reset’ prompts for the annotation buffer even if one is already set. Annotation is entered in a temporary buffer and the annotation is inserted into the working buffer when complete.

**emacspeak-clipboard-copy** (*START END &optional PROMPT*) [Interactive Command]

*control e control c*

Copy contents of the region to the emacspeak clipboard. Previous contents of the clipboard will be overwritten. The Emacspeak clipboard is a convenient way of sharing information between independent Emacspeak sessions running on the same or different machines. Do not use this for sharing information within an Emacs session – Emacs’ register commands are far more efficient and light-weight. Optional interactive prefix arg results in Emacspeak prompting for the clipboard file to use. Argument *START* and *END* specifies region. Optional argument *PROMPT* specifies whether we prompt for the name of a clipboard file.

**emacspeak-clipboard-paste** (**&optional** *PASTE-TABLE*) [Interactive Command]

*control e control y*

Yank contents of the Emacspeak clipboard at point. The Emacspeak clipboard is a convenient way of sharing information between independent Emacspeak sessions running on the same or different machines. Do not use this for sharing information within an Emacs session – Emacs’ register commands are far more efficient and light-weight. Optional interactive prefix arg pastes from the emacspeak table clipboard instead.

**emacspeak-copy-current-file** () [Interactive Command]

*control e meta c*

Copy file visited in current buffer to new location. Prompts for the new location and preserves modification time when copying. If location is a directory, the file is copied to that directory under its current name ; if location names a file in an existing directory, the specified name is used. Asks for confirmation if the copy will result in an existing file being overwritten.

**emacspeak-curl** (*URL*) [Interactive Command]

Grab URL using Curl, and preview it with a browser .

**emacspeak-customize** () [Interactive Command]

*control e cap C*

Customize Emacspeak.

**emacspeak-customize-personal-settings** (*FILE*) [Interactive Command]

Create a customization buffer for browsing and updating personal customizations.

**emacspeak-cvs-berlios-get-project-snapshot** (*PROJECT &optional MODULE*) [Interactive Command]

Grab CVS snapshot of specified project from berlios.de. Ask for module name if prefix argument is given

**emacspeak-cvs-get-anonymous** () [Interactive Command]

Get latest cvs snapshot of emacspeak.

- emacspeak-cvs-gnu-get-project-snapshot** (*PROJECT* [Interactive Command]  
**&optional MODULE**)  
 Grab CVS snapshot of specified project from gnu. Ask for module name if prefix argument is given
- emacspeak-cvs-sf-get-project-snapshot** (*PROJECT* [Interactive Command]  
**&optional MODULE**)  
 Grab CVS snapshot of specified project from sf. Ask for module name if prefix argument is given
- emacspeak-emergency-tts-restart** () [Interactive Command]  
*control e <insert>*  
 For use in an emergency. Will start TTS engine specified by emacspeak-emergency-tts-server.
- emacspeak-frame-label-or-switch-to-labelled-frame** (**&optional** [Interactive Command]  
**PREFIX**)  
*control e meta f*  
 Switch to labelled frame. With optional PREFIX argument, label current frame.
- emacspeak-frame-read-frame-label** () [Interactive Command]  
 Read a frame label with completion.
- emacspeak-generate-documentation** (*FILENAME*) [Interactive Command]  
 Generate docs for all emacspeak commands. Prompts for FILENAME in which to save the documentation. Warning! Contents of file filename will be overwritten.
- emacspeak-generate-texinfo-command-documentation** [Interactive Command]  
 (*FILENAME*)  
 Generate texinfo documentation for all emacspeak commands into file commands.texi. Warning! Contents of file commands.texi will be overwritten.
- emacspeak-generate-texinfo-option-documentation** [Interactive Command]  
 (*FILENAME*)  
 Generate texinfo documentation for all emacspeak options into file filename. Warning! Contents of file filename will be overwritten.
- emacspeak-kill-buffer-quietly** () [Interactive Command]  
 Kill current buffer without asking for confirmation.
- emacspeak-learn-emacs-mode** () [Interactive Command]  
*control e <f1> control h control l <f1> control l <help> control l*  
 Helps you learn the keys. You can press keys and hear what they do. To leave, press C-g.
- emacspeak-link-current-file** () [Interactive Command]  
*control e meta l*  
 Link (hard link) file visited in current buffer to new location. Prompts for the new location and preserves modification time when linking. If location is a directory, the

file is copied to that directory under its current name ; if location names a file in an existing directory, the specified name is used. Signals an error if target already exists.

- emacspeak-links** (*URL*) [Interactive Command]  
 Launch links on specified URL in a new terminal.
- emacspeak-lynx** (*URL*) [Interactive Command]  
 Launch lynx on specified URL in a new terminal.
- emacspeak-next-frame-or-buffer** (**&optional** *FRAME*) [Interactive Command]  
*<control right>*  
 Move to next buffer. With optional interactive prefix arg ‘frame’, move to next frame instead.
- emacspeak-previous-frame-or-buffer** (**&optional** *FRAME*) [Interactive Command]  
*<control left>*  
 Move to previous buffer. With optional interactive prefix arg ‘frame’, move to previous frame instead.
- emacspeak-root** (**&optional** *CD*) [Interactive Command]  
*control e control r*  
 Start a root shell or switch to one that already exists. Optional interactive prefix arg ‘cd’ executes cd default-directory after switching.
- emacspeak-select-this-buffer-next-display** () [Interactive Command]  
*control e <control right>*  
 Select this buffer as displayed in a ‘next’ frame. See documentation for command ‘emacspeak-select-this-buffer-other-window-display’ for the meaning of ‘next’.
- emacspeak-select-this-buffer-other-window-display** (**&optional** *ARG*) [Interactive Command]  
 Switch to this buffer as displayed in a different frame. Emacs allows you to display the same buffer in multiple windows or frames. These different windows can display different portions of the buffer. This is equivalent to leaving a book open at places at once. This command allows you to move to the places where you have left the book open. The number used to invoke this command specifies which of the displays you wish to select. Typically you will have two or at most three such displays open. The current display is 0, the next is 1, and so on. Optional argument *ARG* specifies the display to select.
- emacspeak-select-this-buffer-previous-display** () [Interactive Command]  
*control e <control left>*  
 Select this buffer as displayed in a ‘previous’ window. See documentation for command ‘emacspeak-select-this-buffer-other-window-display’ for the meaning of ‘previous’.

- emacspeak-show-personality-at-point** () [Interactive Command]  
*control e meta v*  
 Show value of property personality (and possibly face) at point.
- emacspeak-show-property-at-point** (**&optional** [Interactive Command]  
*PROPERTY*)  
 Show value of PROPERTY at point. If optional arg property is not supplied, read it interactively. Provides completion based on properties that are of interest. If no property is set, show a message and exit.
- emacspeak-skip-blank-lines-backward** () [Interactive Command]  
*<shift> <up>*  
 Move backward across blank lines. The line under point is then spoken. Signals beginning of buffer.
- emacspeak-skip-blank-lines-forward** () [Interactive Command]  
*<shift> <down>*  
 Move forward across blank lines. The line under point is then spoken. Signals end of buffer.
- emacspeak-speak-browse-linux-howto** (*HOWTO*) [Interactive Command]  
*control e cap H*  
 Browse a Linux Howto file. We cleanup underlining, and set up outline mode correctly.
- emacspeak-speak-hostname** () [Interactive Command]  
*control e meta h*  
 Speak host name.
- emacspeak-speak-popup-messages** () [Interactive Command]  
*control h cap M <f1> cap M <help> cap M*  
 Pop up messages buffer. If it is already selected then hide it and try to restore previous window configuration.
- emacspeak-speak-run-shell-command** (*COMMAND* [Interactive Command]  
**&optional** *AS-ROOT*)  
*control e !*  
 Invoke shell *COMMAND* and display its output as a table. The results are placed in a buffer in Emacspeak's table browsing mode. Optional interactive prefix arg *as-root* runs the command as root. Use this for running shell commands that produce tabulated output. This command should be used for shell commands that produce tabulated output that works with Emacspeak's table recognizer. Verify this first by running the command in a shell and executing command 'emacspeak-table-display-table-in-region' normally bound to C-e TAB.
- emacspeak-speak-show-active-network-interfaces** [Interactive Command]  
**(&optional** *ADDRESS*)  
*control e cap I*

Shows all active network interfaces in the echo area. With interactive prefix argument ADDRESS it prompts for a specific interface and shows its address. The address is also copied to the kill ring for convenient yanking.

`emacspeak-speak-show-memory-used` () [Interactive Command]  
Convenience command to view state of memory used in this session so far.

`emacspeak-speak-telephone-directory` (&optional EDIT) [Interactive Command]  
Lookup and display a phone number. With prefix arg, opens the phone book for editing.

`emacspeak-speak-this-buffer-next-display` () [Interactive Command]  
*control e <right>*

Speak this buffer as displayed in a ‘previous’ window. See documentation for command ‘emacspeak-speak-this-buffer-other-window-display’ for the meaning of ‘next’.

`emacspeak-speak-this-buffer-other-window-display` [Interactive Command]  
(&optional ARG)  
*control e /*

Speak this buffer as displayed in a different frame. Emacs allows you to display the same buffer in multiple windows or frames. These different windows can display different portions of the buffer. This is equivalent to leaving a book open at places at once. This command allows you to listen to the places where you have left the book open. The number used to invoke this command specifies which of the displays you wish to speak. Typically you will have two or at most three such displays open. The current display is 0, the next is 1, and so on. Optional argument ARG specifies the display to speak.

`emacspeak-speak-this-buffer-previous-display` () [Interactive Command]  
*control e <aleft>*

Speak this buffer as displayed in a ‘previous’ window. See documentation for command ‘emacspeak-speak-this-buffer-other-window-display’ for the meaning of ‘previous’.

`emacspeak-ssh-tts-restart` () [Interactive Command]  
*control e <delete>*  
Restart specified ssh tts server.

`emacspeak-sudo` (COMMAND) [Interactive Command]  
SUDO command –run command as super user.

`emacspeak-switch-to-previous-buffer` () [Interactive Command]  
Switch to most recently used interesting buffer. Obsoleted by ‘previous-buffer’ in Emacs 22

`emacspeak-symlink-current-file` () [Interactive Command]  
*control e meta s*

Link (symbolic link) file visited in current buffer to new location. Prompts for the new location and preserves modification time when linking. If location is a directory, the file is copied to that directory under its current name ; if location names a file in an existing directory, the specified name is used. Signals an error if target already exists.

- `emacspeak-view-emacspeak-doc` () [Interactive Command]  
 Display a summary of all Emacspeak commands.
- `emacspeak-view-emacspeak-faq` () [Interactive Command]  
*control e cap F*  
 Browse the Emacspeak FAQ.
- `emacspeak-view-emacspeak-news` () [Interactive Command]  
*control e cap N*  
 Display info on recent change to Emacspeak.
- `emacspeak-view-emacspeak-tips` () [Interactive Command]  
*control e cap T*  
 Browse Emacspeak productivity tips.
- `emacspeak-wizards-add-autoload-cookies` (**&optional** *F*) [Interactive Command]  
 Add autoload cookies to file *f*. Default is to add autoload cookies to current file.
- `emacspeak-wizards-count-slides-in-region` (*START* *END*) [Interactive Command]  
 Count slides starting from point.
- `emacspeak-wizards-dvi-display` () [Interactive Command]  
 Called to set up preview of an DVI file. Assumes we are in a buffer visiting a .DVI file. Previews those contents as text and nukes the buffer visiting the DVI file.
- `emacspeak-wizards-dvi-mode` () [Interactive Command]  
 Major mode for browsing DVI files.  
 DVI files are converted to text and previewed using text mode.  
 This mode runs the hook ‘emacspeak-wizards-dvi-mode-hook’, as the final step during initialization.  
 key binding — ——
- `emacspeak-wizards-edit-file-as-root` (*FILENAME*) [Interactive Command]  
 Edit file as root using sudo vi. See */etc/sudoers* for how to set up sudo.
- `emacspeak-wizards-find-emacspeak-source` () [Interactive Command]  
 Like find-file, but binds default-directory to emacspeak-directory.
- `emacspeak-wizards-find-grep` (*GLOB PATTERN*) [Interactive Command]  
 Run compile using find and grep. Interactive arguments specify filename pattern and search pattern.

- emacspeak-wizards-find-longest-line-in-region** [Interactive Command]  
*(START END)*  
 Find longest line in region. Moves to the longest line when called interactively.
- emacspeak-wizards-find-longest-paragraph-in-region** [Interactive Command]  
*(START END)*  
 Find longest paragraph in region. Moves to the longest paragraph when called interactively.
- emacspeak-wizards-find-shortest-line-in-region** [Interactive Command]  
*(START END)*  
 Find shortest line in region. Moves to the shortest line when called interactively.
- emacspeak-wizards-finder-find** *(DIRECTORY)* [Interactive Command]  
 Run find-dired on specified switches after prompting for the directory to where find is to be launched.
- emacspeak-wizards-finder-mode** () [Interactive Command]  
 Emacspeak Finder  
 This mode runs the hook ‘emacspeak-wizards-finder-mode-hook’, as the final step during initialization.  
 key binding — ——
- emacspeak-wizards-fix-read-only-text** *(START END)* [Interactive Command]  
 Nuke read-only property on text range.
- emacspeak-wizards-fix-typo** *(EXT WORD CORRECTION)* [Interactive Command]  
 Search and replace recursively in all files with extension ‘ext’ for ‘word’ and replace it with correction. Use with caution.
- emacspeak-wizards-generate-finder** () [Interactive Command]  
 Generate a widget-enabled finder wizard.
- emacspeak-wizards-generate-voice-sampler** *(STEP)* [Interactive Command]  
 Generate a buffer that shows a sample line in all the ACSS settings for the current voice family.
- emacspeak-wizards-get-table-content-from-file** [Interactive Command]  
*(TASK FILE DEPTH COUNT)*  
 Extract table specified by depth and count from HTML content at file. Extracted content is placed as a csv file in task.csv.
- emacspeak-wizards-get-table-content-from-url** [Interactive Command]  
*(TASK URL DEPTH COUNT)*  
 Extract table specified by depth and count from HTML content at URL. Extracted content is placed as a csv file in task.csv.
- emacspeak-wizards-google-hits** () [Interactive Command]  
 Filter Google results after performing search to show just the hits.

- `emacspeak-wizards-google-transcode ()` [Interactive Command]  
View Web through Google Transcoder.
- `emacspeak-wizards-how-many-matches (START END &optional PREFIX)` [Interactive Command]  
If you define a file local variable called ‘emacspeak-occur-pattern’ that holds a regular expression that matches lines of interest, you can use this command to conveniently run ‘how-many’ to count matching header lines. With interactive prefix arg, prompts for and remembers the file local pattern.
- `emacspeak-wizards-i810-display-status ()` [Interactive Command]  
Show display status on thinkpads using i810switch.
- `emacspeak-wizards-move-and-speak (COMMAND COUNT)` [Interactive Command]  
Speaks a chunk of text bounded by point and a target position. Target position is specified using a navigation command and a count that specifies how many times to execute that command first. Point is left at the target position. Interactively, command is specified by pressing the key that invokes the command.
- `emacspeak-wizards-occur-header-lines (PREFIX)` [Interactive Command]  
If you define a file local variable called ‘emacspeak-occur-pattern’ that holds a regular expression that matches header lines, you can use this command to conveniently run ‘occur’ to find matching header lines. With prefix arg, prompts for and sets value of the file local pattern.
- `emacspeak-wizards-pdf-open (FILENAME)` [Interactive Command]  
Open pdf file as text.
- `emacspeak-wizards-popup-input-buffer (MODE)` [Interactive Command]  
Provide an input buffer in a specified mode.
- `emacspeak-wizards-portfolio-quotes ()` [Interactive Command]  
Bring up detailed stock quotes for portfolio specified by emacspeak-wizards-personal-portfolio.
- `emacspeak-wizards-ppp-toggle ()` [Interactive Command]  
Bring up or bring down ppp.
- `emacspeak-wizards-ppt-display ()` [Interactive Command]  
Called to set up preview of an PPT file. Assumes we are in a buffer visiting a .ppt file. Previews those contents as HTML and nukes the buffer visiting the ppt file.
- `emacspeak-wizards-ppt-mode ()` [Interactive Command]  
Major mode for browsing PPT slides.  
PPT files are converted to HTML and previewed using a browser.  
In addition to any hooks its parent mode ‘text-mode’ might have run, this mode runs the hook ‘emacspeak-wizards-ppt-mode-hook’, as the final step during initialization.  
key binding — ——

- `emacspeak-wizards-refresh-shell-history` () [Interactive Command]  
Refresh shell history from disk. This is for use in conjunction with bash to allow multiple emacs shell buffers to share history information.
- `emacspeak-wizards-rivo` (*WHEN CHANNEL STOP-TIME OUTPUT DIRECTORY*) [Interactive Command]  
Rivo wizard. Prompts for relevant information and schedules a rivo job using UNIX AT scheduling facility. RIVO is implemented by `rivo.pl` — a Perl script that can be used to launch streaming media and record streaming media for a specified duration.
- `emacspeak-wizards-rpm-query-in-dired` () [Interactive Command]  
Run `rpm -qi` on current dired entry.
- `emacspeak-wizards-shell-toggle` () [Interactive Command]  
*control e <f11>*  
Switch to the shell buffer and `cd` to the directory of the current buffer.
- `emacspeak-wizards-show-commentary` (**&optional FILE**) [Interactive Command]  
Display commentary. Default is to display commentary from current buffer.
- `emacspeak-wizards-show-environment-variable` (*V*) [Interactive Command]  
Display value of specified environment variable.
- `emacspeak-wizards-show-eval-result` (*FORM*) [Interactive Command]  
*control e &*  
Convenience command to pretty-print and view Lisp evaluation results.
- `emacspeak-wizards-show-face` (*FACE*) [Interactive Command]  
Show salient properties of specified face.
- `emacspeak-wizards-speak-iso-datetime` (*ISO*) [Interactive Command]  
Make ISO date-time speech friendly.
- `emacspeak-wizards-spot-words` (*EXT WORD*) [Interactive Command]  
Searches recursively in all files with extension ‘ext’ for ‘word’ and displays hits in a compilation buffer.
- `emacspeak-wizards-squeeze-blanks` (*START END*) [Interactive Command]  
Squeeze multiple blank lines in current buffer.
- `emacspeak-wizards-terminal` (*PROGRAM*) [Interactive Command]  
Launch terminal and rename buffer appropriately.
- `emacspeak-wizards-toggle-mm-dd-yyyy-date-pronouncer` () [Interactive Command]  
Toggle pronunciation of mm-dd-yyyy dates.
- `emacspeak-wizards-tpctl-display-status` () [Interactive Command]  
Show display status on thinkpads using `tpctl`.

|                                                                                                                                                                         |                       |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-wizards-tramp-open-location</code> ( <i>NAME</i> )                                                                                                      | [Interactive Command] |
| Open specified tramp location. Location is specified by name.                                                                                                           |                       |
| <code>emacspeak-wizards-unhex-uri</code> ( <i>URI</i> )                                                                                                                 | [Interactive Command] |
| UnEscape URI                                                                                                                                                            |                       |
| <code>emacspeak-wizards-units</code> ()                                                                                                                                 | [Interactive Command] |
| Run units in a comint sub-process.                                                                                                                                      |                       |
| <code>emacspeak-wizards-use-w3-or-w3m</code> ()                                                                                                                         | [Interactive Command] |
| Alternates between using W3 and W3M for browse-url.                                                                                                                     |                       |
| <code>emacspeak-wizards-vc-n</code> ()                                                                                                                                  | [Interactive Command] |
| Accelerator for VC viewer.                                                                                                                                              |                       |
| <code>emacspeak-wizards-vc-viewer</code> ( <i>CONSOLE</i> )                                                                                                             | [Interactive Command] |
| View contents of specified virtual console.                                                                                                                             |                       |
| <code>emacspeak-wizards-vc-viewer-mode</code> ()                                                                                                                        | [Interactive Command] |
| Major mode for interactively viewing virtual console contents.                                                                                                          |                       |
| key binding — ——                                                                                                                                                        |                       |
| C-l emacspeak-wizards-vc-viewer-refresh                                                                                                                                 |                       |
| This mode runs the hook ‘emacspeak-wizards-vc-viewer-mode-hook’, as the final step during initialization.                                                               |                       |
| <code>emacspeak-wizards-vc-viewer-refresh</code> ()                                                                                                                     | [Interactive Command] |
| Refresh view of VC we’re viewing.                                                                                                                                       |                       |
| <code>emacspeak-wizards-vi-as-su-file</code> ( <i>FILE</i> )                                                                                                            | [Interactive Command] |
| Launch sudo vi on specified file in a terminal.                                                                                                                         |                       |
| <code>emacspeak-wizards-voice-sampler</code> ( <i>PERSONALITY</i> )                                                                                                     | [Interactive Command] |
| Read a personality and apply it to the current line.                                                                                                                    |                       |
| <code>emacspeak-wizards-vpn-toggle</code> ()                                                                                                                            | [Interactive Command] |
| Bring up or bring down vpn.                                                                                                                                             |                       |
| <code>emacspeak-wizards-xl-display</code> ()                                                                                                                            | [Interactive Command] |
| Called to set up preview of an XL file. Assumes we are in a buffer visiting a .xls file. Previews those contents as HTML and nukes the buffer visiting the xls file.    |                       |
| <code>emacspeak-wizards-xl-mode</code> ()                                                                                                                               | [Interactive Command] |
| Major mode for browsing XL spreadsheets.                                                                                                                                |                       |
| XL Sheets are converted to HTML and previewed using a browser.                                                                                                          |                       |
| In addition to any hooks its parent mode ‘text-mode’ might have run, this mode runs the hook ‘emacspeak-wizards-xl-mode-hook’, as the final step during initialization. |                       |
| key binding — ——                                                                                                                                                        |                       |

## 11.68 emacspeak-xml-shell

Automatically generated documentation for commands defined in module emacspeak-xml-shell.

|                                                                                                                                                                                             |                       |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-xml-shell</code> ( <i>SYSTEM-ID</i> )                                                                                                                                       | [Interactive Command] |
| Start Xml-Shell on contents of system-id.                                                                                                                                                   |                       |
| <code>emacspeak-xml-shell-browse-current</code> ()                                                                                                                                          | [Interactive Command] |
| Display current node.                                                                                                                                                                       |                       |
| <code>emacspeak-xml-shell-browse-result</code> ( <i>XPATH</i> )                                                                                                                             | [Interactive Command] |
| Display XPath and display its result using W3.                                                                                                                                              |                       |
| <code>emacspeak-xml-shell-goto-children</code> ()                                                                                                                                           | [Interactive Command] |
| Navigate down to the children of current node.                                                                                                                                              |                       |
| <code>emacspeak-xml-shell-goto-next-child</code> ()                                                                                                                                         | [Interactive Command] |
| Navigate forward to the next child of current node.                                                                                                                                         |                       |
| <code>emacspeak-xml-shell-goto-parent</code> ()                                                                                                                                             | [Interactive Command] |
| Navigate up to the parent of current node.                                                                                                                                                  |                       |
| <code>emacspeak-xml-shell-goto-previous-child</code> ()                                                                                                                                     | [Interactive Command] |
| Navigate backward to the previous child of current node.                                                                                                                                    |                       |
| <code>emacspeak-xml-shell-mode</code> ()                                                                                                                                                    | [Interactive Command] |
| XML Shell                                                                                                                                                                                   |                       |
| Interactive XML browser. key binding — ——                                                                                                                                                   |                       |
| C-c Prefix Command <down> emacspeak-xml-shell-goto-children <left> emacspeak-xml-shell-goto-previous-child <right> emacspeak-xml-shell-goto-next-child <up> emacspeak-xml-shell-goto-parent |                       |
| C-c C-v emacspeak-xml-shell-browse-result C-c v emacspeak-xml-shell-browse-current                                                                                                          |                       |
| In addition to any hooks its parent mode ‘comint-mode’ might have run, this mode runs the hook ‘emacspeak-xml-shell-mode-hook’, as the final step during initialization.                    |                       |

## 11.69 emacspeak-xslt

Automatically generated documentation for commands defined in module emacspeak-xslt.

|                                                                                                         |                       |
|---------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-xslt-view</code> ( <i>STYLE URL</i> )                                                   | [Interactive Command] |
| Browse URL with specified XSL style.                                                                    |                       |
| <code>emacspeak-xslt-view-region</code> ( <i>STYLE START END</i><br>&optional <i>UNESCAPE-CHARENT</i> ) | [Interactive Command] |
| Browse XML region with specified XSL style.                                                             |                       |
| <code>emacspeak-xslt-view-xml</code> ( <i>STYLE URL</i> &optional <i>UNESCAPE-CHARENT</i> )             | [Interactive Command] |
| Browse XML URL with specified XSL style.                                                                |                       |

## 11.70 emacspeak-zinf

Commentary: Defines a simple derived mode for interacting with zinf. zinf == zinf is not freeamp zinf navigation commands then work via single keystrokes.

Automatically generated documentation for commands defined in module emacspeak-zinf.

**emacspeak-zinf** (*RESOURCE*) [Interactive Command]

Play specified resource using zinf. Resource is an MP3 file or m3u playlist. The player is placed in a buffer in emacspeak-zinf-mode.

**emacspeak-zinf-mode** () [Interactive Command]

Major mode for zinf interaction.

key binding — ——

+ emacspeak-zinf-zinf-call-command - emacspeak-zinf-zinf-call-command  
 = emacspeak-zinf-zinf-call-command ? emacspeak-zinf-zinf-call-command  
 B emacspeak-zinf-zinf-call-command F emacspeak-zinf-zinf-call-command  
 J emacspeak-zinf-zinf-call-command b emacspeak-zinf-zinf-call-command f  
 emacspeak-zinf-zinf-call-command j emacspeak-zinf-zinf-call-command p ..  
 q emacspeak-zinf-zinf-call-command s emacspeak-zinf-zinf-call-command z  
 emacspeak-zinf <left> emacspeak-aumix-wave-decrease <right> emacspeak-aumix-  
 wave-increase

This mode runs the hook ‘emacspeak-zinf-mode-hook’, as the final step during initialization.

**emacspeak-zinf-zinf-call-command** () [Interactive Command]

Call appropriate zinf command.

**emacspeak-zinf-zinf-command** (*CHAR*) [Interactive Command]

Execute Zinf command.

## 11.71 emacspeak

Commentary: The complete audio desktop.

Emacspeak extends Emacs to be a fully functional audio desktop. This is the main emacspeak module. It actually does very little: It loads the various parts of the system.

Automatically generated documentation for commands defined in module emacspeak.

**emacspeak-describe-emacspeak** () [Interactive Command]

*control h control e <f1> control e <help> control e*

Give a brief overview of emacspeak.

**emacspeak-submit-bug** () [Interactive Command]

*control e CONTROL meta b*

Function to submit a bug to the programs maintainer.

## 11.72 voice-setup

Commentary: A voice is to audio as a font is to a visual display. A personality is to audio as a face is to a visual display.

Voice-lock-mode is a minor mode that causes your comments to be spoken in one personality, strings in another, reserved words in another, documentation strings in another, and so on.

Comments will be spoken in ‘emacspeak-voice-lock-comment-personality’. Strings will be spoken in ‘emacspeak-voice-lock-string-personality’. Doc strings will be spoken in ‘emacspeak-voice-lock-doc-string-personality’. Function and variable names (in their defining forms) will be spoken in ‘emacspeak-voice-lock-function-name-personality’. Reserved words will be spoken in ‘emacspeak-voice-lock-keyword-personality’.

To make the text you type be voiceified, use M-x voice-lock-mode. When this minor mode is on, the voices of the current line are updated with every insertion or deletion.

How faces map to voices: TTS engine specific modules e.g., dectalk-voices.el and outloud-voices.el define a standard set of voice names. This module maps standard "personality" names to these pre-defined voices. It does this via special form def-voice-font which takes a personality name, a voice name and a face name to set up the mapping between face and personality, and personality and voice. Newer Emacspeak modules should use voice-setup-add-map when defining face->personality mappings. Older code calls def-voice-font directly, but over time those calls will be changed to the more succinct form provided by voice-setup-add-map. For use from other modules, also see function voice-setup-map-face which is useful when mapping a single face. Both voice-setup-add-map and voice-setup-map-face call special form def-voice-font.

Special form def-voice-font sets up the personality name to be available via custom. new voices can be defined using CSS style specifications see special form defvoice Voices defined via defvoice can be customized via custom see the documentation for defvoice.

Automatically generated documentation for commands defined in module voice-setup.

**voice-setup-list-voices** (*PATTERN*) [Interactive Command]

Show all defined voice-face mappings in a help buffer. Sample text to use comes from variable ‘voice-setup-sample-text’.

**voice-setup-toggle-silence-personality** () [Interactive Command]

*control e meta q*

Toggle audibility of personality under point . If personality at point is currently audible, its face->personality map is cached in a buffer local variable, and its face->personality map is replaced by face->inaudible. If personality at point is inaudible, and there is a cached value, then the original face->personality mapping is restored. In either case, the buffer is refontified to have the new mapping take effect.

## 11.73 emacspeak-moz

Commentary: MozRepl provides a read-eval-print loop into Firefox This module provides convenient functions for driving MozRepl See <http://repo.hyperstruct.net/mozlab>

Automatically generated documentation for commands defined in module emacspeak-moz.

|                                                                                                                                     |                       |
|-------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| <code>emacspeak-moz-browse-current ()</code><br>Browse current node.<br>(fn)                                                        | [Interactive Command] |
| <code>emacspeak-moz-browser-back ()</code><br>Move back in history.<br>(fn)                                                         | [Interactive Command] |
| <code>emacspeak-moz-browser-forward ()</code><br>Move forward in history.<br>(fn)                                                   | [Interactive Command] |
| <code>emacspeak-moz-close-tab-or-browser ()</code><br>Close tab, or browser when one tab left.<br>(fn)                              | [Interactive Command] |
| <code>emacspeak-moz-down ()</code><br>Go Down a level and browse.<br>(fn)                                                           | [Interactive Command] |
| <code>emacspeak-moz-eval-expression-and-browse ()</code><br>Send expression to Moz, get output, and browse it in Emacs.<br>(fn EXP) | [Interactive Command] |
| <code>emacspeak-moz-eval-expression-and-go ()</code><br>Send expression to Moz and switch to it.<br>(fn EXP)                        | [Interactive Command] |
| <code>emacspeak-moz-filter-and-browse ()</code><br>Browse document filtered by XPath filter.<br>(fn XPATH)                          | [Interactive Command] |
| <code>emacspeak-moz-goto-url ()</code><br>Make Firefox used by our repl Go to the specified URL.<br>(fn URL)                        | [Interactive Command] |
| <code>emacspeak-moz-goto-url-at-point ()</code><br>Make Firefox used by our repl Go to url under point.<br>(fn)                     | [Interactive Command] |
| <code>emacspeak-moz-inspect ()</code><br>Inspect specified object.<br>(fn WHAT)                                                     | [Interactive Command] |
| <code>emacspeak-moz-jump ()</code><br>Jump to specified index in history.<br>(fn INDEX)                                             | [Interactive Command] |

|                                                             |                       |
|-------------------------------------------------------------|-----------------------|
| <code>emacspeak-moz-left ()</code>                          | [Interactive Command] |
| Go left and browse.<br>(fn)                                 |                       |
| <code>emacspeak-moz-refresh ()</code>                       | [Interactive Command] |
| Reload document.<br>(fn)                                    |                       |
| <code>emacspeak-moz-right ()</code>                         | [Interactive Command] |
| Go right and browse.<br>(fn)                                |                       |
| <code>emacspeak-moz-search ()</code>                        | [Interactive Command] |
| Search for pattern in current context.<br>(fn PATTERN)      |                       |
| <code>emacspeak-moz-up ()</code>                            | [Interactive Command] |
| Go Up a level and browse.<br>(fn)                           |                       |
| <code>emacspeak-moz-visit-next-and-browse ()</code>         | [Interactive Command] |
| Asks visitor to go forward and browses the result.<br>(fn)  |                       |
| <code>emacspeak-moz-visit-previous-and-browse ()</code>     | [Interactive Command] |
| Asks visitor to go backward and browses the result.<br>(fn) |                       |

### 11.74 emacspeak-org

Commentary: Speech-enable org — Org allows you to keep organized notes and todo lists. Homepage: <http://www.astro.uva.nl/~dominik/Tools/org/>

Automatically generated documentation for commands defined in module emacspeak-org.

|                                           |                       |
|-------------------------------------------|-----------------------|
| <code>emacspeak-org-popup-input ()</code> | [Interactive Command] |
| Pops up an org input area.<br>(fn)        |                       |

### 11.75 emacspeak-webmarks

Commentary: <http://www.google.com/bookmarks> provides a simple bookmark facility emacspeak-webmarks provides direct minibuffer-level access to these in the spirit of the Emacspeak Web Command Line. For now, the emacspeak-webmarks-key needs to be set by hand after doing the following: 0) sign in to google using xml-forms/glogin.xml 1) search for a bookmark using xml-forms/bookmark-find.xml will work around this eventually.

Automatically generated documentation for commands defined in module emacspeak-webmarks.

|                                            |                       |
|--------------------------------------------|-----------------------|
| <code>emacspeak-webmarks-add ()</code>     | [Interactive Command] |
| Add WebMark.<br>(fn URL TITLE NOTES)       |                       |
| <code>emacspeak-webmarks-history ()</code> | [Interactive Command] |
| Search search history.<br>(fn QUERY)       |                       |
| <code>emacspeak-webmarks-list ()</code>    | [Interactive Command] |
| List WebMarks.<br>(fn)                     |                       |
| <code>emacspeak-webmarks-search ()</code>  | [Interactive Command] |
| Search WebMarks.<br>(fn QUERY)             |                       |

## 11.76 URL Templates

This section is generated automatically from the source-level documentation. Any errors or corrections should be made to the source-level documentation. This section documents a total of 130 URL Templates.

All of these URL templates can be invoked via command *M-x emacspeak-url-template-fetch* normally bound to *control e u*. This command prompts for the name of the template, and completion is available via Emacs' minibuffer completion. Each URL template carries out the following steps:

- Prompt for the relevant information.
- Fetch the resulting URL using an appropriate fetcher.
- Set up the resulting resource with appropriate customizations.

As an example, the URL templates that enable access to NPR media streams prompt for a program id and date, and automatically launch the realmedia player after fetching the resource.

### *Airport conditions*

Display airport conditions from the FAA.

### *All Things Considered Stream from NPR*

Play NPR All Things Considered stream.

### *All Things Considered from NPR*

Play All Things Considered segment.

### *Amazon Product Details By ASIN*

Retrieve product details from Amazon by either ISBN or ASIN.

### *American Life On Demand.*

Play This American Life shows on demand.

### *Anonymize Google Search*

Logout from Google to do an anonymous search.

*Answers.com*

Search answers.com

*Archive News Search*

Search Google Archive News.

*Ask Local Search*

Ask Local Search.

*Ask Walking Directions*

Walking directions from Ask.com

*BBC 7 Schedule*

Retrieve BBC7 schedule for specified day.

*BBC Channel On Demand*

Display BBC Channel on demand.

*BBC Genres On Demand*

Display BBC Channel on demand.

*BBC Listen Again*

BBC Listen Again Listings

*BBC News*

BBC News text version.

*BBC Programs On Demand*

Play BBC programs on demand.

*BBC Radio4 On Demand*

Specify a week day (three letters – lower case – and a time spec – e.g. 1230 – to play a BBC Radio4 program on demand.

*BBC Radio7 On Demand*

Specify a week day (three letters – lower case – and a time spec – e.g. 1230 – to play a BBC Radio7 program on demand.

*BBC Sports*

BBC News text version.

*Baseball Game Index*

Display baseball Play By Play.

*Baseball Play By Play*

Display baseball Play By Play.

*Baseball scores*

Display baseball scores.

*Baseball standings*

Display MLB standings.

*BlogSearch Google*

Google Blog Search

*BookShare*

Bookshare Login

*CNN Content*

CNN Content

*CNN Market Data*

CNN Money

*CNN Market News*

CNN Money

*CNN Technology*

CNN Technology news.

*CNN headlines*

Retrieve and speak headline news from CNN.

*CNNContent*

Filter down to CNN content area.

*CNNFn Content*

Extract content links from Money.

*CNNPodCasts*

List CNN Podcast media links.

*Cartoon You Said It By Laxman*

Retrieve Cartoon Times Of India.

*Code Search From Google*

Search using Google Code Search.

*DDJ TechNetCast Play*

Play Technetcast stream from DDJ.

*DDJ TechNetCast Save*

Browse to a specified DDJ Technetcast stream and save it.

*Dictionary Lookup*

Dictionary Lookup

*Digg*

Display Digg Feed.

*EmacsWiki Search*

EmacsWiki Search

*EmapSpeak Via Google*

EmapSpeak Via Google.

Specify the query using English and addresses as complete as possible.

Here are some examples:

0) To find a location by address specify:

650 Harry Road San Jose CA 95120

1) To get directions, specify:

<source address> to <destination address>

2) To find businesses etc., near a location, specify:

<what> near <location address>

*Finance Google Search*

Display content from Google Finance.

*Flight Tracker*

Display flight arrival and departure information.

*GCalendar Mobile*

Google Calendar XHTML version.

*GMail Mobile*

GMail Mobile XHTML version — light-weight, fast!

*Geek Linux Daily*

Play specified edition of Geek Linux DailyShow

*Google Atom News*

Display specified news feed.

*Google Books*

Google Books Search with text when available

*Google Feeds*

List Google news Feeds.

*Google Glossary*

Google Glossary lookup.

*Google Hits*

Only show Google hits.

*Google Image Search*

Google Image Search

*Google Local*

Google Local Search.

*Google Mobile Search*

Google Mobile Search

*Google News Search*

Search Google news.

*Google Print*

Google Print Search

*Google RSS News*

Search Google news.

*Google Reader*

View feed via Google Reader.

*Google Recent News Search*

Search Google news.

*Google Scholar*

Google Scholar Search

*Google Soccer Results*

Display World Cup Soccer Results for specified countries.

*Google Text News*

Retrieve and speak Google News Overview.

*Google TimeLine View*

Do a Google search and get a timeline view of results.

*Google Transcoder*

Transcode site via Google.

*Google Video*

Retrieve Google Video search results as an atom feed.

*Google WebQuotes*

Google WebQuotes.

*InfoWorld RSS Feeds*

Produce a set of RSS links published by InfoWorld.

*Linux Today News*

Get news column from Linux Today.

*MDC Search*

Search MDC Wiki.

*MLB Scorecard*

Show MLB Scorecard.

*MapQuest Directions*

Retrieve and speak directions from MapQuest.

*Meerkat Profile*

Meerkat Profile

*Meerkat Recipe*

Meerkat tool

*Morning Edition Stream from NPR*

Play NPR Morning Edition stream.

*Morning Edition from NPR*

Play Morning Edition segment.

*Motley Fool Radio from NPR*

Play NPR Motley Fool stream.

*Mozilla Developers Corner*

Pull specified resource from MDC.

*Multilingual dictionary via Google.*

Translate word using Google. Source and target languages are specified as two-letter language codes, e.g. en|de translates from English to German

*Music Search*

Music search on Google.

*NPR On Demand*

Play NPR shows on demand. Program is specified as a program code:

ME Morning Edition ATC All Things Considered day Day To Day newsnotes News  
And Notes totn Talk Of The Nation fa Fresh Air wesat Weekend Edition Saturday wesun  
Weekend Edition Sunday fool The Motley Fool

Segment is specified as a two digit number –specifying a blank value plays entire program.

*NY Times RSS Feeds*

Display browsable list of NY Times RSS Feeds.

*Netcraft Site Report*

Analyze WWW site using Netcraft.

*Netcraft Web Analysis*

Analyze WWW site using Netcraft.

*Official GoogleBlog Search*

Search within all official Google blogs.

*Old Time Radio*

This months Old Time Radio Programing

*On The Fly CSE*

Build a CSE on the fly and use it to search.

*Patent Search From Google*

Perform patent search via Google

*Periodicals from Bookshare*

Fetch periodical from Bookshare.

*Pulpit --- I Cringely*

Read pulpit from PBS. Published on the Thursday of the week.

*Reader Subscription Search*

Search within feeds subscribed to in Google Reader.

*Redhat Linux Show*

Play specified edition of Redhat Linux Show

*Reuters Finance*

Reuters Finance Lookup

*Shoutcast Search*

Locate and display Shoutcast streams.

*Talk Of The Nation Stream from NPR*

Play NPR Talk Of The Nation stream.

*Talk Of The Nation from NPR*

Play NPR Talk Of The Nation segment.

*Tech News From CNet*

Display tech news from CNET

*Times Of India*

Retrieve Times Of India. Set up URL rewrite rule to get print page.

*Translation Via Google*

Translate a Web page using google. Source and target languages are specified as two-letter language codes, e.g. en|de translates from English to German.

*Travelocity Lookup*

Show arrival/departure information from Travelocity.

*UPS Packages*

Display package tracking information from UPS.

*Wait Wait, Dont Tell Me (NPR)*

Play Wait, Wait Dont Tell Me from NPR.

*Weather forecast from Weather Underground*

Weather forecast from weather underground mobile.

*WordNet Search*

Look up term in WordNet.

*Yahoo Business News*

Retrieve and speak business section from Yahoo Daily News.

*Yahoo Content By Content ID*

Retrieve and speak news section from Yahoo Daily News.

*Yahoo DailyNews*

Retrieve and speak DailyNewspage from Yahoo Daily News.

*Yahoo Entertainment*

Retrieve and speak Entertainment section from Yahoo Daily News.

*Yahoo Health*

Retrieve and speak Health section from Yahoo Daily News.

*Yahoo Lifestyle*

Yahoo Lifestyle News.

*Yahoo Oddly*

Retrieve and speak Oddity section from Yahoo Daily News.

*Yahoo Politics*

Retrieve and speak Politics section from Yahoo Daily News.

*Yahoo RSS Feeds*

List Yahoo RSS Feeds.

*Yahoo SF Local*

Retrieve and speak Local section from Yahoo Daily News.

*Yahoo Science*

Retrieve and speak Science section from Yahoo Daily News.

*Yahoo Sports*

Entertainment news from Yahoo.

*Yahoo Technology News*

Yahoo Technology News.

*Yahoo Top Stories*

Retrieve and speak Top Stories section from Yahoo Daily News.

*Yahoo World News*

Yahoo World News.

*YubNub Web Command Line*

YubNub Command Line

*ask search mobile*

Mobile search using Ask.com

*fedex packages*

Display package tracking information from Fedex.

*mlb standings*

Get XML feed containing team standings.

*pdf2html*

Use access.adobe.com to convert a remote PDF document to plain text. The PDF document needs to be available on the public Internet.

*pdf2txt*

Use access.adobe.com to convert a remote PDF document to plain text. The PDF document needs to be available on the public Internet.

*rss weather from wunderground*

Pull RSS weather feed for specified state/city.

*sourceforge Download*

Download specified file.

*sourceforge browse mirrors*

Retrieve download page at Sourceforge for specified project.

*sourceforge project*

Open specified project page at SourceForge.

*w3c IRC Logs*

Use this to pull up the archived logs from the W3C IRC. You need to know the exact name of the channel.

*w3c Lists*

Use this to pull up the archived mail from the W3C list. You need to know the exact name of the list.

## 12 Emacspeak Customizations

This chapter is generated automatically from the source-level documentation. Any errors or corrections should be made to the source-level documentation. This chapter documents a total of 639 user customizable options.

### 12.1 cd-tool Options

Commentary: Provide an emacs front-end to cdtool. cdtool can be obtained as an rpm check using rpmfind or from its home site at sunsite.unc.edu /pub/Linux/apps/sound/cdrom/cli This module also provides the ability to play or save clips from a CD if you have cdda2wav installed. cdda2wav is a cd to wav convertor.

Automatically generated documentation for options defined in module cd-tool. These options are customizable via Emacs' Custom interface.

**User Option** *cd-tool-start-command* [Variable]  
 \*Name of cdstart command; most likely either "cdstart" or "cdplay".

### 12.2 dtk-speak Options

Commentary: Defines the TTS interface.

Automatically generated documentation for options defined in module dtk-speak. These options are customizable via Emacs' Custom interface.

**User Option** *dtk-cleanup-patterns* [Variable]  
 List of repeating patterns to clean up. You can use command 'dtk-add-cleanup-pattern' bound to C-e d a to add more patterns. Specify patterns that people use to decorate their ASCII files, and cause untold pain to the speech synthesizer.  
 If more than 3 consecutive occurrences of a specified pattern is found, the TTS engine replaces it with a repeat count.

**User Option** *dtk-resume-should-toggle* [Variable]  
 \*T means 'dtk-resume' acts as a toggle.

**User Option** *dtk-speak-nonprinting-chars* [Variable]  
 \*Option that specifies handling of non-printing chars. Non nil value means non printing characters should be spoken as their octal value. Set this to t to avoid a dectalk bug that makes the speech box die if it seems some accented characters in certain contexts.

**User Option** *dtk-speech-rate-base* [Variable]  
 \*Value of lowest tolerable speech rate.

**User Option** *dtk-speech-rate-step* [Variable]  
 \*Value of speech rate increment. This determines step size used when setting speech rate via command 'dtk-set-predefined-speech-rate'. Formula used is dtk-speech-rate-base + dtk-speech-rate-step\*level.

**User Option** *dtk-startup-hook* [Variable]  
List of hooks to be run after starting up the speech server. Set things like speech rate, punctuation mode etc in this hook.

**User Option** *dtk-stop-immediately-while-typing* [Variable]  
\*Set it to nil if you dont want speech to flush as you type. You can use command 'dtk-toggle-stop-immediately-while-typing' bound to C-e d I to toggle this setting.

**User Option** *dtk-use-tones* [Variable]  
Allow tones to be turned off.

**User Option** *tts-strip-octals* [Variable]  
Set to T to strip all octal chars before speaking. Particularly useful for web browsing.

### 12.3 emacspeak-add-log Options

Commentary: speech-enables change-log-mode Code:

```
(require 'cl) (declaim (optimize (safety 0) (speed 3))) (require 'custom) (require
'emacspeak-preamble)
(defgroup emacspeak-add-log nil "Customize Emacspeak for change-log-mode and
friends." :group 'emacspeak)
(voice-setup-add-map '((change-log-acknowledgement voice-smoothen) (change-log-
conditionals voice-animate) (change-log-email voice-womanize-1) (change-log-function
voice-bolden-extra) (change-log-file voice-bolden) (change-log-email voice-womanize-1)
(change-log-list voice-lighten) (change-log-name voice-lighten-extra)))
(provide 'emacspeak-add-log)
local variables: folded-file: t byte-compile-dynamic: t
```

Automatically generated documentation for options defined in module emacspeak-add-log. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-change-log-acknowledgement-voice* [Variable]  
Personality used for change-log-acknowledgement This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.

**User Option** *emacspeak-change-log-conditionals-voice* [Variable]  
Personality used for change-log-conditionals This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.

**User Option** *emacspeak-change-log-email-voice* [Variable]  
Personality used for change-log-email This personality uses voice-womanize-1 whose effect can be changed globally by customizing voice-womanize-1-settings.

**User Option** *emacspeak-change-log-file-voice* [Variable]  
Personality used for change-log-file This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

**User Option** *emacspeak-change-log-function-voice* [Variable]  
 Personality used for change-log-function This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.

**User Option** *emacspeak-change-log-list-voice* [Variable]  
 Personality used for change-log-list This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.

**User Option** *emacspeak-change-log-name-voice* [Variable]  
 Personality used for change-log-name This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.

## 12.4 emacspeak-advice Options

Commentary: This module defines the advice forms for making the core of Emacs speak Advice forms that are specific to Emacs subsystems do not belong here! I violate this at present by advising completion comint and shell here.

Note that we needed to advice a lot more for Emacs 19 and Emacs 20 than we do for Emacs 21 and Emacs 22. As of August 2007, this file is being purged of advice forms not needed in Emacs 22. This also means that this and subsequent versions of Emacspeak should not be run on versions of Emacs older than Emacs 21, And preferably only run on Emacs 22. This version of Emacspeak is only tested on Emacs 22.

Automatically generated documentation for options defined in module emacspeak-advice. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-speak-cue-errors* [Variable]  
 Specifies if error messages are cued.

**User Option** *emacspeak-speak-messages-pause* [Variable]  
 \* Option to make messages pause speech. If t then all messages will pause ongoing speech if any before the message is spoken.

## 12.5 emacspeak-alsaplayer Options

Commentary: Defines a simple derived mode for interacting with alsaplayer. alsaplayer navigation commands work via single keystrokes.

Automatically generated documentation for options defined in module emacspeak-alsaplayer. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-alsaplayer-auditory-feedback* [Variable]  
 Turn this on if you want spoken feedback and auditory icons from alsaplayer.

**User Option** *emacspeak-alsaplayer-media-directory* [Variable]  
 Directory to look for media files.

**User Option** *emacspeak-alsaplayer-program* [Variable]  
 Alsaplayer executable.

## 12.6 emacspeak-amphetadesk Options

Automatically generated documentation for options defined in module emacspeak-amphetadesk. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-amphetadesk-port* [Variable]  
Port where AmphetaDesk listens.

**User Option** *emacspeak-amphetadesk-program* [Variable]  
Script that launches amphetadesk.

**User Option** *emacspeak-amphetadesk-uri* [Variable]  
URI for Amphetadesk home.

## 12.7 emacspeak-aumix Options

Commentary: Provides an AUI to setting up the auditory display via AUMIX This module is presently Linux specific

Automatically generated documentation for options defined in module emacspeak-aumix. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-alsactl-program* [Variable]  
ALSA sound controller used to restore settings.

**User Option** *emacspeak-aumix-full-duplex-p* [Variable]  
\*Set to T if the sound card is truly full duplex.

**User Option** *emacspeak-aumix-multichannel-capable-p* [Variable]  
\*Set to T if the sound card is capable of mixing multiple channels of audio.

**User Option** *emacspeak-aumix-reset-options* [Variable]  
\*Option to pass to aumix for resetting to default values.

**User Option** *emacspeak-aumix-settings-file* [Variable]  
\*Name of file containing personal aumix settings.

## 12.8 emacspeak-calendar Options

Automatically generated documentation for options defined in module emacspeak-calendar. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-calendar-mark-personality* [Variable]  
Personality to use when showing marked calendar entries.

## 12.9 emacspeak-compile Options

Commentary: This module makes compiling code from inside Emacs speech friendly. It is an example of how a little amount of code can make Emacspeak even better.

Automatically generated documentation for options defined in module emacspeak-compile. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-compilation-column-number-voice* [Variable]  
 Personality used for compilation-column-number This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.

**User Option** *emacspeak-compilation-error-voice* [Variable]  
 Personality used for compilation-error This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.

**User Option** *emacspeak-compilation-info-voice* [Variable]  
 Personality used for compilation-info This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.

**User Option** *emacspeak-compilation-line-number-voice* [Variable]  
 Personality used for compilation-line-number This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.

**User Option** *emacspeak-compilation-warning-voice* [Variable]  
 Personality used for compilation-warning This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.

## 12.10 emacspeak-custom Options

Automatically generated documentation for options defined in module emacspeak-custom. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-custom-group-regexp* [Variable]  
 Pattern identifying start of custom group.

**User Option** *emacspeak-custom-toolbar-regexp* [Variable]  
 Pattern that identifies toolbar section.

## 12.11 emacspeak-daisy Options

Commentary: Daisy Digital Talking Book Reader

Automatically generated documentation for options defined in module emacspeak-daisy. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-daisy-books-directory* [Variable]  
 Customize this to the root of where books are organized.

**User Option** *emacspeak-daisy-completion-extensions-to-ignore* [Variable]  
 These file name extensions are ignored when locating the navigation file for a book. Include all extensions except '.ncx' for optimal performance.

## 12.12 emacspeak-dired Options

Commentary: This module speech enables dired. It reduces the amount of speech you hear: Typically you hear the file names as you move through the dired buffer. Voicification is used to indicate directories, marked files etc.

Automatically generated documentation for options defined in module emacspeak-dired. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-dired-file-cmd-options* [Variable]  
Options passed to Unix builtin 'file' command.

## 12.13 emacspeak-dismal Options

Automatically generated documentation for options defined in module emacspeak-dismal. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-dismal-value-personality* [Variable]  
Personality used for speaking cell values in summaries.

## 12.14 emacspeak-ediff Options

Automatically generated documentation for options defined in module emacspeak-ediff. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-ediff-A-personality* [Variable]  
Personality used to voiceify difference chunk A

**User Option** *emacspeak-ediff-B-personality* [Variable]  
Personality used to voiceify difference chunk B

**User Option** *emacspeak-ediff-always-autorefine-diffs* [Variable]  
Says if emacspeak should try computing fine differences each time. Set this to nil if things get too slow.

**User Option** *emacspeak-ediff-fine-A-personality* [Variable]  
Personality used to voiceify difference chunk A

**User Option** *emacspeak-ediff-fine-B-personality* [Variable]  
Personality used to voiceify difference chunk B

## 12.15 emacspeak-entertain Options

Commentary: Auditory interface to misc games

Automatically generated documentation for options defined in module emacspeak-entertain. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-mpuz-solved-personality* [Variable]  
Personality used for mpuz-solved-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.

**User Option** *emacspeak-mpuz-trivial-personality* [Variable]  
 Personality used for mpuz-trivial-face This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.

**User Option** *emacspeak-mpuz-unsolved-personality* [Variable]  
 Personality used for mpuz-unsolved-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

## 12.16 emacspeak-eperiodic Options

Automatically generated documentation for options defined in module emacspeak-eperiodic. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-eperiodic-media-location* [Variable]  
 Location of streaming media describing elements.

## 12.17 emacspeak-erc Options

Commentary: erc.el is a modern Emacs client for IRC including color and font locking support. erc.el - an Emacs IRC client (by Alexander L. Belikoff)  
<http://www.cs.cmu.edu/~berez/irc/erc.el>

Automatically generated documentation for options defined in module emacspeak-erc. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-erc-ignore-notices* [Variable]  
 Set to T if you dont want to see notification messages from the server.

**User Option** *emacspeak-erc-my-nick* [Variable]  
 My IRC nick.

**User Option** *emacspeak-erc-speak-all-participants* [Variable]  
 Speak all things said if t.

## 12.18 emacspeak-eshell Options

Commentary: EShell is a shell implemented entirely in Emacs Lisp. It is part of emacs 21 –and can also be used under Emacs 20. This module speech-enables EShell

```
Play an auditory icon as you display the prompt (defun emacspeak-eshell-prompt-
function () "Play auditory icon for prompt." (declare (special eshell-last-command-
status)) (cond ((= 0 eshell-last-command-status) (emacspeak-serve-auditory-icon 'item))
(t (emacspeak-auditory-icon 'warn-user))))
```

```
(add-hook 'eshell-after-prompt-hook 'emacspeak-eshell-prompt-function)
```

```
Speak command output (add-hook 'eshell-post-command-hook (function (lambda nil
(declare (special eshell-last-input-end eshell-last-output-end eshell-last-output-start))
(emacspeak-speak-region eshell-last-input-end eshell-last-output-end))) t)
```

```
(defadvice pcomplete-list (after emacspeak pre act) "Provide auditory feedback."
(when (interactive-p) (emacspeak-auditory-icon 'help) (emacspeak-auditory-icon 'help)))
```

```
(defadvice pcomplete-show-completions (around emacspeak pre act comp) (let
((emacspeak-speak-messages nil)) ad-do-it))
```

```
(defadvice eshell (after emacspeak pre act) "Announce switching to shell mode. Provide an auditory icon if possible." (when (interactive-p) (emacspeak-auditory-icon 'select-object) (emacspeak-setup-programming-mode) (emacspeak-dtk-sync) (emacspeak-speak-line)))
```

```
(loop for f in '(eshell-next-input eshell-previous-input eshell-next-matching-input eshell-previous-matching-input eshell-next-matching-input-from-input eshell-previous-matching-input-from-input) do (eval ('(defadvice (, f) (after emacspeak pre act comp) "Speak selected command." (when (interactive-p) (emacspeak-auditory-icon 'select-object) (save-excursion (beginning-of-line) (eshell-skip-prompt) (emacspeak-speak-line 1)))))))
```

```
(defgroup emacspeak-eshell nil "EShell on the Emacspeak Audio Desktop." :group 'emacspeak :group 'eshell :prefix "emacspeak-eshell-")
```

```
(defcustom emacspeak-eshell-ls-use-personalities t "Indicates if ls in eshell uses different voice personalities." :type 'boolean :group 'emacspeak-eshell)
```

```
(voice-setup-add-map '((eshell-ls-archive voice-lighten-extra) (eshell-ls-archive-face voice-lighten-extra) (eshell-ls-backup voice-monotone-medium) (eshell-ls-backup-face voice-monotone-medium) (eshell-ls-clutter voice-smoothen-extra) (eshell-ls-clutter-face voice-smoothen-extra) (eshell-ls-directory voice-bolden) (eshell-ls-directory-face voice-bolden) (eshell-ls-executable voice-animate-extra) (eshell-ls-executable-face voice-animate-extra) (eshell-ls-missing voice-brighten) (eshell-ls-missing-face voice-brighten) (eshell-ls-product voice-lighten-medium) (eshell-ls-product-face voice-lighten-medium) (eshell-ls-readonly voice-monotone) (eshell-ls-readonly-face voice-monotone) (eshell-ls-special voice-lighten-extra) (eshell-ls-special-face voice-lighten-extra) (eshell-ls-symlink voice-smoothen) (eshell-ls-symlink-face voice-smoothen) (eshell-ls-unreadable voice-monotone-medium) (eshell-ls-unreadable-face voice-monotone-medium) (eshell-prompt voice-animate) (eshell-prompt-face voice-animate)))
```

```
(loop for f in '(eshell-next-prompt eshell-previous-prompt eshell-forward-matching-input eshell-backward-matching-input) do (eval ('(defadvice (, f) (after emacspeak pre act comp) "Speak selected command." (when (interactive-p) (let ((emacspeak-speak-messages nil)) (emacspeak-auditory-icon 'select-object) (emacspeak-speak-line 1)))))))
```

```
(loop for f in '(eshell-insert-buffer-name eshell-insert-process eshell-insert-envvar) do (eval ('(defadvice (, f) (after emacspeak pre act comp) "Speak output." (when (interactive-p) (emacspeak-auditory-icon 'select-object) (emacspeak-speak-line))))))
```

```
(defadvice eshell-insert-process (after emacspeak pre act comp) "Speak output." (when (interactive-p) (emacspeak-auditory-icon 'select-object) (emacspeak-speak-line))) (defadvice eshell-delchar-or-maybe-eof (around emacspeak pre act) "Speak character you're deleting." (cond ((interactive-p) (cond ((= (point) (point-max)) (message "Sending EOF to comint process")) (t (dtk-tone 500 30 'force) (emacspeak-speak-char t))) ad-do-it) (t ad-do-it)) ad-return-value)
```

```
(defadvice eshell-delete-backward-char (around emacspeak pre act) "Speak character you're deleting." (cond ((interactive-p) (dtk-tone 500 30 'force) (emacspeak-speak-this-char (preceding-char))) ad-do-it) (t ad-do-it)) ad-return-value)
```

```
(defadvice eshell-show-output (after emacspeak pre act comp) "Speak output." (when (interactive-p) (let ((emacspeak-show-point t) (voice-lock-mode t)) (emacspeak-auditory-icon 'large-movement) (emacspeak-speak-region (point) (mark)))) (defadvice
```

```
eshell-mark-output (after emacspeak pre act comp) "Speak output." (when (interactive-p)
(let ((emacspeak-show-point t) (voice-lock-mode t)) (emacspeak-auditory-icon
'mark-object) (emacspeak-speak-line)))) (defadvice eshell-kill-output (after emacspeak pre
act comp) "Produce auditory feedback." (when (interactive-p) (emacspeak-auditory-icon
'delete-object) (message "Flushed output")))
```

```
(defadvice eshell-kill-input (before emacspeak pre act) "Provide spoken feedback."
(when (interactive-p) (emacspeak-auditory-icon 'delete-object) (emacspeak-speak-line)))
```

```
(defadvice eshell-toggle (after emacspeak pre act comp) "Provide spoken context
feedback." (when (interactive-p) (cond ((eq major-mode 'eshell-mode) (emacspeak-
setup-programming-mode) (emacspeak-speak-line)) (t (emacspeak-speak-mode-line)))
(emacspeak-auditory-icon 'select-object))) (defadvice eshell-toggle-cd (after emacspeak
pre act comp) "Provide spoken context feedback." (when (interactive-p) (cond ((eq
major-mode 'eshell-mode) (emacspeak-speak-line)) (t (emacspeak-speak-mode-line)))
(emacspeak-auditory-icon 'select-object)))
```

```
(provide 'emacspeak-eshell)
```

```
local variables: folded-file: t byte-compile-dynamic: t
```

Automatically generated documentation for options defined in module emacspeak-eshell. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-eshell-ls-use-personalities* [Variable]  
Indicates if ls in eshell uses different voice personalities.

## 12.19 emacspeak-eterm Options

Commentary: This module makes eterm talk. Eterm is the new terminal emulator for Emacs. Use of emacspeak with eterm really needs an info page. At present, the only documentation is the source level documentation. This module uses Control-t as an additional prefix key to allow the user To move around the terminal and have different parts spoken.

Automatically generated documentation for options defined in module emacspeak-eterm. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-eterm-bold-personality* [Variable]  
Personality to indicate terminal bold.

**User Option** *emacspeak-eterm-default-personality* [Variable]  
Default personality for terminal.

**User Option** *emacspeak-eterm-highlight-personality* [Variable]  
Personality to show terminal highlighting.

**User Option** *emacspeak-eterm-remote-hosts-cache* [Variable]  
File where list of known remote hosts is cached

**User Option** *emacspeak-eterm-underline-personality* [Variable]  
Personality to indicate terminal underlining.

## 12.20 emacspeak-forms Options

Automatically generated documentation for options defined in module emacspeak-forms. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-forms-ro-voice* [Variable]  
Personality for read-only fields.

**User Option** *emacspeak-forms-rw-voice* [Variable]  
Personality for read-write fields.

## 12.21 emacspeak-gnus Options

Automatically generated documentation for options defined in module emacspeak-gnus. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-gnus-punctuation-mode* [Variable]  
Pronunciation mode to use for gnus buffers.

## 12.22 emacspeak-hide Options

Commentary: Flexible hide and show for emacspeak. This module allows one to easily hide or expose blocks of lines starting with a common prefix. It is motivated by the need to flexibly hide quoted text in email but is designed to be more general. the prefix parsing is inspired by filladapt.el

Automatically generated documentation for options defined in module emacspeak-hide. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-hidden-header-line-personality* [Variable]  
Personality used to identify header lines of blocks.

## 12.23 emacspeak-ido Options

Commentary: speech-enable ido.el This is an interesting task since most of the value-add provided by package ido.el is visual feedback. Speech UI Challenge: What is the most efficient means of conveying a dynamically updating set of choices? current strategy is to walk the list using c-s and c-r as provided by ido Set number matches shown to 3 using Custom so you dont hear the entire list.

Automatically generated documentation for options defined in module emacspeak-ido. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-ido-first-match-voice* [Variable]  
Personality used for ido-first-match This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.

**User Option** *emacspeak-ido-incomplete-regexp-voice* [Variable]  
Personality used for ido-incomplete-regexp This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.

**User Option** *emacspeak-ido-indicator-voice* [Variable]  
Personality used for ido-indicator This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.

**User Option** *emacspeak-ido-only-match-voice* [Variable]  
 Personality used for ido-only-match This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

**User Option** *emacspeak-ido-subdir-voice* [Variable]  
 Personality used for ido-subdir This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.

## 12.24 emacspeak-imenu Options

Automatically generated documentation for options defined in module emacspeak-imenu. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-imenu-autospeak* [Variable]  
 Speak contents of sections automatically if set.

## 12.25 emacspeak-info Options

Automatically generated documentation for options defined in module emacspeak-info. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-info-select-node-speak-chunk* [Variable]  
 \*Specifies how much of the selected node gets spoken. Possible values are: screen-full – speak the displayed screen node – speak the entire node.

## 12.26 emacspeak-ispell Options

Automatically generated documentation for options defined in module emacspeak-ispell. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-ispell-max-choices* [Variable]  
 Emacspeak will not speak the choices if there are more than this many available corrections.

## 12.27 emacspeak-jabber Options

Commentary: emacs-jabber.el implements a jabber client for emacs emacs-jabber is hosted at sourceforge. I use emacs-jabber with my gmail.com account

Automatically generated documentation for options defined in module emacspeak-jabber. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-jabber-activity-personality* [Variable]  
 Personality used for jabber-activity-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.

**User Option** *emacspeak-jabber-chat-error-voice* [Variable]  
 Personality used for jabber-chat-error This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.

- User Option** *emacspeak-jabber-chat-prompt-foreign-voice* [Variable]  
Personality used for jabber-chat-prompt-foreign This personality uses voice-brighten-medium whose effect can be changed globally by customizing voice-brighten-medium-settings.
- User Option** *emacspeak-jabber-chat-prompt-local-voice* [Variable]  
Personality used for jabber-chat-prompt-local This personality uses voice-smoothen-medium whose effect can be changed globally by customizing voice-smoothen-medium-settings.
- User Option** *emacspeak-jabber-chat-prompt-system-voice* [Variable]  
Personality used for jabber-chat-prompt-system This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.
- User Option** *emacspeak-jabber-chat-text-foreign-voice* [Variable]  
Personality used for jabber-chat-text-foreign This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-jabber-chat-text-local-voice* [Variable]  
Personality used for jabber-chat-text-local This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-jabber-rare-time-personality* [Variable]  
Personality used for jabber-rare-time-face This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.
- User Option** *emacspeak-jabber-roster-user-away-voice* [Variable]  
Personality used for jabber-roster-user-away This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-jabber-roster-user-chatty-voice* [Variable]  
Personality used for jabber-roster-user-chatty This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-jabber-roster-user-dnd-voice* [Variable]  
Personality used for jabber-roster-user-dnd This personality uses voice-lighten-medium whose effect can be changed globally by customizing voice-lighten-medium-settings.
- User Option** *emacspeak-jabber-roster-user-error-voice* [Variable]  
Personality used for jabber-roster-user-error This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.
- User Option** *emacspeak-jabber-roster-user-offline-voice* [Variable]  
Personality used for jabber-roster-user-offline This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.

**User Option** *emacspeak-jabber-roster-user-online-voice* [Variable]  
 Personality used for jabber-roster-user-online This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

**User Option** *emacspeak-jabber-roster-user-xa-voice* [Variable]  
 Personality used for jabber-roster-user-xa This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.

**User Option** *emacspeak-jabber-speak-presence-alerts* [Variable]  
 Set to T if you want to hear presence alerts.

**User Option** *emacspeak-jabber-title-large-voice* [Variable]  
 Personality used for jabber-title-large This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.

**User Option** *emacspeak-jabber-title-medium-voice* [Variable]  
 Personality used for jabber-title-medium This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

**User Option** *emacspeak-jabber-title-small-voice* [Variable]  
 Personality used for jabber-title-small This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.

## 12.28 emacspeak-keymap Options

Automatically generated documentation for options defined in module emacspeak-keymap. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-alt-keys* [Variable]  
 \*Specifies alt key bindings for the audio desktop. You can turn the 'Pause' key on your Linux PC keyboard into a 'alt' key on Linux by having it emit the sequence 'C-x@a'.

Bindings specified here are available on prefix key 'alt' (not to be confused with alt==meta) for example, if you bind 's' to command emacspeak-emergency-tts-restart then that command will be available on key 'ALT s'

The value of this variable is an association list. The car of each element specifies a key sequence. The cdr specifies an interactive command that the key sequence executes. To enter a key with a modifier, type C-q followed by the desired modified keystroke. For example, to enter C-s (Control s) as the key to be bound, type C-q C-s in the key field in the customization buffer. You can use the notation [f1], [f2], etc., to specify function keys.

**User Option** *emacspeak-hyper-keys* [Variable]  
 \*Specifies hyper key bindings for the audio desktop. Emacs can use the 'hyper' key as a modifier key. You can turn the 'windows' keys on your Linux PC keyboard into a 'hyper' key on Linux by having it emit the sequence 'C-x@h'.

Bindings specified here are available on prefix key 'hyper' for example, if you bind 'b' to command 'bbdb ' then that command will be available on key 'hyper b'.

The value of this variable is an association list. The car of each element specifies a key sequence. The cdr specifies an interactive command that the key sequence executes. To enter a key with a modifier, type C-q followed by the desired modified keystroke. For example, to enter C-s (Control s) as the key to be bound, type C-q C-s in the key field in the customization buffer. You can use the notation [f1], [f2], etc., to specify function keys.

**User Option** *emacspeak-personal-keys* [Variable]

\*Specifies personal key bindings for the audio desktop. Bindings specified here are available on prefix key C-e x for example, if you bind ‘s’ to command `emacspeak-emergency-tts-restart` then that command will be available on key C-e x s

The value of this variable is an association list. The car of each element specifies a key sequence. The cdr specifies an interactive command that the key sequence executes. To enter a key with a modifier, type C-q followed by the desired modified keystroke. For example, to enter C-s (Control s) as the key to be bound, type C-q C-s in the key field in the customization buffer. You can use the notation [f1], [f2], etc., to specify function keys.

**User Option** *emacspeak-super-keys* [Variable]

\*Specifies super key bindings for the audio desktop. You can turn the right ‘windows menu’ keys on your Linux PC keyboard into a ‘super’ key on Linux by having it emit the sequence ‘C-x@s’.

Bindings specified here are available on prefix key ‘super’ for example, if you bind ‘s’ to command `emacspeak-emergency-tts-restart` then that command will be available on key ‘super s’

The value of this variable is an association list. The car of each element specifies a key sequence. The cdr specifies an interactive command that the key sequence executes. To enter a key with a modifier, type C-q followed by the desired modified keystroke. For example, to enter C-s (Control s) as the key to be bound, type C-q C-s in the key field in the customization buffer. You can use the notation [f1], [f2], etc., to specify function keys.

## 12.29 emacspeak-loaddefs Options

Automatically generated documentation for options defined in module `emacspeak-loaddefs`. These options are customizable via Emacs’ Custom interface.

**User Option** *emacspeak-character-echo* [Variable]

If t, then emacspeak echoes characters as you type. You can use C-e d k to toggle this setting.

**User Option** *emacspeak-line-echo* [Variable]

If t, then emacspeak echoes lines as you type. You can use C-e d l to set this option.

**User Option** *emacspeak-play-program* [Variable]

Name of executable that plays sound files.

**User Option** *emacspeak-sounds-default-theme* [Variable]  
 Default theme for auditory icons.

**User Option** *emacspeak-speak-embedded-url-pattern* [Variable]  
 Pattern to recognize embedded URLs.

**User Option** *emacspeak-word-echo* [Variable]  
 If t, then emacspeak echoes words as you type. You can use C-e d w to toggle this option.

### 12.30 emacspeak-m-player Options

Commentary: Defines a simple derived mode for interacting with mplayer. mplayer is a versatile media player capable of playing many streaming formats and is especially useful for playing windows media (WMA) and streaming windows media (ASF) files. mplayer is available on the WWW: RPM package <http://mirrors.sctpc.com/dominik/linux/pkgs/mplayer/i586/mplayer-0.90pre5-2.i586.rpm> You may need the win32 codecs which can be downloaded from <http://ftp.lug.udel.edu/MPlayer/releases/w32codec-0.60.tar.bz2> Mplayer FAQ at <http://www.mplayerhq.hu/DOCS/faq.html> Mplayer docs at <http://www.mplayerhq.hu/DOCS/>

Automatically generated documentation for options defined in module emacspeak-m-player. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-m-player-options* [Variable]  
 Options passed to mplayer.

**User Option** *emacspeak-m-player-program* [Variable]  
 Media player program.

### 12.31 emacspeak-madplay Options

Commentary: Defines a simple derived mode for interacting with madplay. madplay navigation commands work via single keystrokes.

Automatically generated documentation for options defined in module emacspeak-madplay. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-madplay-media-directory* [Variable]  
 Directory to look for media files.

**User Option** *emacspeak-madplay-program* [Variable]  
 Script to invoke madplay. Emacspeak comes with a sample amadplay script in etc/amadplay that sets up madplay to pipe output to als.

### 12.32 emacspeak-make-mode Options

Commentary: This module speech enables make-mode

Automatically generated documentation for options defined in module emacspeak-make-mode. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-makefile-makepp-perl-voice* [Variable]  
 Personality used for makefile-makepp-perl This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.

**User Option** *emacspeak-makefile-shell-voice* [Variable]  
 Personality used for makefile-shell This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.

**User Option** *emacspeak-makefile-space-voice* [Variable]  
 Personality used for makefile-space This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.

**User Option** *emacspeak-makefile-targets-voice* [Variable]  
 Personality used for makefile-targets This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

### 12.33 emacspeak-message Options

Automatically generated documentation for options defined in module emacspeak-message. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-message-punctuation-mode* [Variable]  
 Pronunciation mode to use for message buffers.

### 12.34 emacspeak-muse Options

Commentary: Speech enable Muse

Automatically generated documentation for options defined in module emacspeak-muse. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-muse-bad-link-personality* [Variable]  
 Personality used for muse-bad-link-face This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.

**User Option** *emacspeak-muse-emphasis-1-voice* [Variable]  
 Personality used for muse-emphasis-1 This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.

**User Option** *emacspeak-muse-emphasis-2-voice* [Variable]  
 Personality used for muse-emphasis-2 This personality uses voice-lighten-medium whose effect can be changed globally by customizing voice-lighten-medium-settings.

**User Option** *emacspeak-muse-emphasis-3-voice* [Variable]  
 Personality used for muse-emphasis-3 This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.

**User Option** *emacspeak-muse-header-1-voice* [Variable]  
 Personality used for muse-header-1 This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

- User Option** *emacspeak-muse-header-2-voice* [Variable]  
 Personality used for muse-header-2 This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-muse-header-3-voice* [Variable]  
 Personality used for muse-header-3 This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-muse-header-4-voice* [Variable]  
 Personality used for muse-header-4 This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-muse-header-5-voice* [Variable]  
 Personality used for muse-header-5 This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-muse-link-personality* [Variable]  
 Personality used for muse-link-face This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-muse-verbatim-personality* [Variable]  
 Personality used for muse-verbatim-face This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.

### 12.35 emacspeak-ocr Options

Commentary: This module defines Emacspeak front-end to OCR. This module assumes that sane is installed and working for image acquisition, and that there is an OCR engine that can take acquired images and produce text. Prerequisites: Sane installed and working. scanimage to generate tiff files from scanner. tiffcp to compress the tiff file. working ocr executable by default this module assumes that the OCR executable is named "ocr"

Automatically generated documentation for options defined in module emacspeak-ocr. These options are customizable via Emacs' Custom interface.

- User Option** *emacspeak-ocr-compress-image* [Variable]  
 Command used to compress the scanned tiff file.
- User Option** *emacspeak-ocr-compress-image-options* [Variable]  
 Options used for compressing tiff image.
- User Option** *emacspeak-ocr-compress-photo-options* [Variable]  
 Options used when created JPEG from scanned photographs.
- User Option** *emacspeak-ocr-engine* [Variable]  
 OCR engine to process acquired image.
- User Option** *emacspeak-ocr-engine-options* [Variable]  
 Command line options to pass to OCR engine.
- User Option** *emacspeak-ocr-image-extension* [Variable]  
 Filename extension used for acquired image.

|                                                                 |            |
|-----------------------------------------------------------------|------------|
| <b>User Option</b> <i>emacspeak-ocr-jpeg-metadata-writer</i>    | [Variable] |
| Program to add metadata to JPEG files.                          |            |
| <b>User Option</b> <i>emacspeak-ocr-keep-uncompressed-image</i> | [Variable] |
| If set to T, uncompressed image is not removed.                 |            |
| <b>User Option</b> <i>emacspeak-ocr-photo-compress</i>          | [Variable] |
| Program to create JPEG compressed images.                       |            |
| <b>User Option</b> <i>emacspeak-ocr-scan-image</i>              | [Variable] |
| Name of image acquisition program.                              |            |
| <b>User Option</b> <i>emacspeak-ocr-scan-image-options</i>      | [Variable] |
| Command line options to pass to image acquisition program.      |            |
| <b>User Option</b> <i>emacspeak-ocr-scan-photo-options</i>      | [Variable] |
| Options used when scanning in photographs.                      |            |
| <b>User Option</b> <i>emacspeak-ocr-working-directory</i>       | [Variable] |
| Directory where images and OCR results will be placed.          |            |

### 12.36 emacspeak-outline Options

Commentary: Provide additional advice to outline-mode

Automatically generated documentation for options defined in module emacspeak-outline. These options are customizable via Emacs' Custom interface.

|                                                                        |            |
|------------------------------------------------------------------------|------------|
| <b>User Option</b> <i>emacspeak-outline-dont-query-before-speaking</i> | [Variable] |
| *Option to control prompts when speaking outline sections.             |            |

### 12.37 emacspeak-pcl-cvs Options

Commentary: Speech-enabled CVS access via package pcl-cvs.el

Automatically generated documentation for options defined in module emacspeak-pcl-cvs. These options are customizable via Emacs' Custom interface.

|                                                                                                                                                                       |            |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>User Option</b> <i>emacspeak-cvs-filename-personality</i>                                                                                                          | [Variable] |
| Personality used for cvs-filename-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.                  |            |
| <b>User Option</b> <i>emacspeak-cvs-handled-personality</i>                                                                                                           | [Variable] |
| Personality used for cvs-handled-face This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings. |            |
| <b>User Option</b> <i>emacspeak-cvs-header-personality</i>                                                                                                            | [Variable] |
| Personality used for cvs-header-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.                    |            |

**User Option** *emacspeak-cvs-marked-personality* [Variable]  
 Personality used for cvs-marked-face This personality uses voice-brighten-medium whose effect can be changed globally by customizing voice-brighten-medium-settings.

**User Option** *emacspeak-cvs-msg-personality* [Variable]  
 Personality used for cvs-msg-face This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings.

**User Option** *emacspeak-cvs-need-action-personality* [Variable]  
 Personality used for cvs-need-action-face This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.

### 12.38 emacspeak-personality Options

Commentary: This module defines a personality interface for implementing voice lock via font lock.

Automatically generated documentation for options defined in module emacspeak-personality. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-personality-show-unmapped-faces* [Variable]  
 If set, faces that dont have a corresponding personality are displayed in the messages area.

**User Option** *emacspeak-personality-voiceify-faces* [Variable]  
 Determines how and if we voiceify faces.  
 None means that faces are not mapped to voices. Prepend means that the corresponding personality is prepended to the existing personalities on the text.  
 Append means place corresponding personality at the end. Simple means that voiceification is not cumulative –this is the default.

**User Option** *emacspeak-personality-voiceify-overlays* [Variable]  
 Determines how and if we voiceify overlays.  
 None means that overlay faces are not mapped to voices. Prepend means that the corresponding personality is prepended to the existing personalities on the text under overlay.  
 Append means place corresponding personality at the end.

### 12.39 emacspeak-pronounce Options

Commentary: This module implements user customizable pronunciation dictionaries for emacspeak. Custom pronunciations can be defined per file, per directory and/or per major mode. Emacspeak maintains a persistent user dictionary upon request and loads these in new emacspeak sessions. This module implements the user interface to the custom dictionary as well as providing the internal API used by the rest of emacspeak in using the dictionary. Algorithm:

The persistent dictionary is a hash table where the hash keys are filenames, directory names, or major-mode names. The hash values are association lists defining the dictionary. Users of this module can retrieve a dictionary made up of all applicable association lists for a given file.

Automatically generated documentation for options defined in module `emacspeak-pronounce`. These options are customizable via Emacs' Custom interface.

**User Option** [Variable]

*emacspeak-pronounce-common-xml-namespace-uri-pronunciations*  
Pronunciations for well known namespace URIs.

**User Option** *emacspeak-pronounce-dictionaries-file* [Variable]

File that holds the persistent emacspeak pronunciation dictionaries.

**User Option** *emacspeak-pronounce-internet-smileys-pronunciations* [Variable]

Pronunciation dictionary used in all instant messenger and IRC chat modes. See <http://oz.uc.edu/~solkode/smileys.html> for a full list.

**User Option** *emacspeak-pronounce-load-pronunciations-on-startup* [Variable]

Says if user dictionaries loaded on emacspeak startup.

**User Option** *emacspeak-pronounce-pronunciation-personality* [Variable]

\*Pronunciation personality. This is the personality used when speaking things that have a pronunciation applied.

## 12.40 emacspeak-re-builder Options

Commentary: Speech-enable re-builder. Will be used to advantage in efficiently setting up outline regexp wizards

Automatically generated documentation for options defined in module `emacspeak-re-builder`. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-reb-match-0-voice* [Variable]

Personality used for reb-match-0 This personality uses voice-lock-overlay-0 whose effect can be changed globally by customizing voice-lock-overlay-0-settings.

**User Option** *emacspeak-reb-match-1-voice* [Variable]

Personality used for reb-match-1 This personality uses voice-lock-overlay-1 whose effect can be changed globally by customizing voice-lock-overlay-1-settings.

**User Option** *emacspeak-reb-match-2-voice* [Variable]

Personality used for reb-match-2 This personality uses voice-lock-overlay-2 whose effect can be changed globally by customizing voice-lock-overlay-2-settings.

**User Option** *emacspeak-reb-match-3-voice* [Variable]

Personality used for reb-match-3 This personality uses voice-lock-overlay-3 whose effect can be changed globally by customizing voice-lock-overlay-3-settings.

## 12.41 emacspeak-realaudio Options

Commentary: Assuming you have a correctly configured RealAudio player, this package provides single click access to starting and stopping a RealAudio stream from anywhere on the Emacspeak desktop. Before using this package, make sure that your realaudio player works outside Emacs. Then set variable Emacspeak-realaudio-player to point to the program you use to play RealAudio streams.

Automatically generated documentation for options defined in module emacspeak-realaudio. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-realaudio-mp3-clipper* [Variable]  
Executable used to clip MP3 files.

**User Option** *emacspeak-realaudio-player* [Variable]  
\*Executable that plays realaudio

**User Option** *emacspeak-realaudio-player-options* [Variable]  
\*Options for realplayer.

**User Option** *emacspeak-realaudio-reset-auditory-display* [Variable]  
Set this to T if you want the audio settings reset after a realaudio stream is done playing.

## 12.42 emacspeak-remote Options

Automatically generated documentation for options defined in module emacspeak-remote. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-remote-hooks* [Variable]  
List of hook functions that are run after emacspeak is set to run as a remote application. Use this to add actions you typically perform after you enter remote mode.

**User Option** *emacspeak-remote-use-ssh* [Variable]  
Set to T to use SSH remote servers.

## 12.43 emacspeak-replace Options

Automatically generated documentation for options defined in module emacspeak-replace. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-replace-personality* [Variable]  
Personality used in search and replace to indicate word that is being replaced.

## 12.44 emacspeak-rpm-spec Options

Automatically generated documentation for options defined in module emacspeak-rpm-spec. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-rpm-spec-dir-personality* [Variable]  
Personality used for rpm-spec-dir-face This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.

**User Option** *emacspeak-rpm-spec-doc-personality* [Variable]  
 Personality used for rpm-spec-doc-face This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.

**User Option** *emacspeak-rpm-spec-ghost-personality* [Variable]  
 Personality used for rpm-spec-ghost-face This personality uses voice-smoothen-medium whose effect can be changed globally by customizing voice-smoothen-medium-settings.

**User Option** *emacspeak-rpm-spec-macro-personality* [Variable]  
 Personality used for rpm-spec-macro-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

**User Option** *emacspeak-rpm-spec-package-personality* [Variable]  
 Personality used for rpm-spec-package-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.

**User Option** *emacspeak-rpm-spec-tag-personality* [Variable]  
 Personality used for rpm-spec-tag-face This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.

## 12.45 emacspeak-sigbegone Options

Commentary: Speech-enables package sigbegone –voiceify sigs in email and news

Automatically generated documentation for options defined in module emacspeak-sigbegone. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-sigbegone-exorcized-personality* [Variable]  
 Personality used for sigbegone-exorcized-face This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.

## 12.46 emacspeak-sounds Options

Commentary: This module provides the interface for generating auditory icons in emacspeak. Design goal: 1) Auditory icons should be used to provide additional feedback, not as a gimmick. 2) The interface should be usable at all times without the icons: e.g. when on a machine without a sound card. 3) General principle for when to use an icon: Convey information about events taking place in parallel. For instance, if making a selection automatically moves the current focus to the next choice, We speak the next choice, while indicating the fact that something was selected with a sound cue. This interface will assume the availability of a shell command "play" that can take one or more sound files and play them. This module will also provide a mapping between names in the elisp world and actual sound files. Modules that wish to use auditory icons should use these names, instead of actual file names. As of Emacspeak 13.0, this module defines a themes architecture for auditory icons. Sound files corresponding to a given theme are found in appropriate subdirectories of emacspeak-sounds-directory

Automatically generated documentation for options defined in module emacspeak-sounds. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-auditory-icon-function* [Variable]  
 \*Function that plays auditory icons.

**User Option** *emacspeak-play-args* [Variable]  
 Set this to -i if using the play program that ships on sunos/solaris. Note: on sparc20's there is a sunos bug that causes the machine to crash if you attempt to play sound when /dev/audio is busy. It's imperative that you use the -i flag to play on sparc20's.

**User Option** *emacspeak-sounds-reset-snd-module-command* [Variable]  
 Command to reset sound module.

## 12.47 emacspeak-speak Options

Commentary: This module defines the core speech services used by emacspeak. It depends on the speech server interface modules. It protects other parts of emacspeak from becoming dependent on the speech server modules.

Automatically generated documentation for options defined in module emacspeak-speak. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-audio-indentation* [Variable]  
 Option indicating if line indentation is cued. If non-nil, then speaking a line indicates its indentation. You can use command 'emacspeak-toggle-audio-indentation' bound to C-e d i to toggle this setting.

**User Option** *emacspeak-audio-indentation-method* [Variable]  
 \*Current technique used to cue indentation. Default is 'speak'. You can specify 'tone' for producing a beep indicating the indentation. Automatically becomes local in any buffer where it is set.

**User Option** *emacspeak-comint-autospeak* [Variable]  
 Says if comint output is automatically spoken. You can use 'emacspeak-toggle-comint-autospeak' bound to C-e C-q to toggle this setting.

**User Option** *emacspeak-decoration-rule* [Variable]  
 \*Regular expressions to match lines that are purely decorative ascii.

**User Option** *emacspeak-horizontal-rule* [Variable]  
 \*Regular expression to match horizontal rules in ascii text.

**User Option** *emacspeak-mail-alert* [Variable]  
 \*Option to indicate cueing of new mail. If t, emacspeak will alert you about newly arrived mail with an auditory icon when displaying the mode line. You can use command 'emacspeak-toggle-mail-alert' bound to C-e M-m to set this option. If you have online access to a voicemail drop, you can have a voice-mail alert set up by specifying the location of the voice-mail drop via custom option emacspeak-voicemail-spool-file.

**User Option** *emacspeak-mail-alert-interval* [Variable]  
 Interval in seconds between mail alerts for the same pending message.

- User Option** *emacspeak-mail-spool-file* [Variable]  
Mail spool file examined to alert you about newly arrived mail.
- User Option** *emacspeak-speak-directory-settings* [Variable]  
\*Name of file that holds directory specific settings.
- User Option** *emacspeak-speak-filter-persistent-store* [Variable]  
File where emacspeak filters are persisted.
- User Option** *emacspeak-speak-line-column-filter* [Variable]  
\*List that specifies columns to be filtered. The list when set holds pairs of start-col.end-col pairs that specifies the columns that should not be spoken. Each column contains a single character –this is inspired by cut -c on UNIX.
- User Option** *emacspeak-speak-line-invert-filter* [Variable]  
Non-nil means the sense of ‘filter’ is inverted when filtering columns in a line –see command *emacspeak-speak-line-set-column-filter*.
- User Option** *emacspeak-speak-load-directory-settings-quietly* [Variable]  
\*User option that affects loading of directory specific settings. If set to T, Emacspeak will not prompt before loading directory specific settings.
- User Option** *emacspeak-speak-maximum-line-length* [Variable]  
\*Threshold for determining ‘long’ lines. Emacspeak will ask for confirmation before speaking lines that are longer than this length. This is to avoid accidentally opening a binary file and torturing the speech synthesizer with a long string of gibberish.
- User Option** *emacspeak-speak-message-again-should-copy-to-kill-ring* [Variable]  
If set, asking for last message will copy it to the kill ring.
- User Option** *emacspeak-speak-messages* [Variable]  
\*Option indicating if messages are spoken. If nil, emacspeak will not speak messages as they are echoed to the message area. You can use command ‘*emacspeak-toggle-speak-messages*’ bound to C-e q.
- User Option** *emacspeak-speak-paragraph-personality* [Variable]  
\*Personality used to mark start of paragraph.
- User Option** *emacspeak-speak-space-regexp* [Variable]  
Pattern that matches white space.
- User Option** *emacspeak-speak-time-format-string* [Variable]  
\*Format string that specifies how the time should be spoken. See the documentation for function ‘*format-time-string*’
- User Option** *emacspeak-speak-zoneinfo-directory* [Variable]  
Directory containing timezone data.
- User Option** *emacspeak-unspeakable-rule* [Variable]  
\*Pattern to match lines of special chars. This is a regular expression that matches lines containing only non-alphanumeric characters. emacspeak will generate a tone instead of speaking such lines when punctuation mode is set to some.

**User Option** *emacspeak-voicemail-spool-file* [Variable]  
Mail spool file examined to alert you about newly arrived voicemail.

## 12.48 emacspeak-speedbar Options

Commentary: This module advises speedbar.el for use with Emacs. The latest speedbar can be obtained from ftp://ftp.ultranet.com/pub/zappo/ This module ensures that speedbar works smoothly outside a windowing system in addition to speech enabling all interactive commands. Emacspeak also adds an Emacspeak environment specific entry point to speedbar –*emacspeak-speedbar-goto-speedbar*– and binds this

Automatically generated documentation for options defined in module *emacspeak-speedbar*. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-speedbar-button-personality* [Variable]  
personality used for speedbar buttons

**User Option** *emacspeak-speedbar-default-personality* [Variable]  
Default personality used in speedbar buffers

**User Option** *emacspeak-speedbar-directory-personality* [Variable]  
Speedbar personality for directory buttons

**User Option** *emacspeak-speedbar-file-personality* [Variable]  
Personality used for file buttons

**User Option** *emacspeak-speedbar-highlight-personality* [Variable]  
Personality used for for speedbar highlight.

**User Option** *emacspeak-speedbar-selected-personality* [Variable]  
Personality used to indicate speedbar selection

**User Option** *emacspeak-speedbar-tag-personality* [Variable]  
Personality used for speedbar tags

## 12.49 emacspeak-table-ui Options

Automatically generated documentation for options defined in module *emacspeak-table-ui*. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-table-column-header-personality* [Variable]  
personality for speaking column headers.

**User Option** *emacspeak-table-row-header-personality* [Variable]  
Personality for speaking row headers

## 12.50 emacspeak-tnt Options

Commentary: Speech-enables TNT – the Emacs AOL Instant Messenger client

Automatically generated documentation for options defined in module *emacspeak-tnt*. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-tnt-autospeak* [Variable]  
True means messages in this chat session will be spoken automatically.

## 12.51 emacspeak-url-template Options

Commentary: It is often useful to have “parameterized hot list entries” i.e., hotlist entries that are “templates” for the actual URL. The user provides values for the parameterized portions of the URL e.g. the date. See Section 11.76 [URL Templates], page 126, for details on the URL templates that are presently defined.

Automatically generated documentation for options defined in module emacspeak-url-template. These options are customizable via Emacs’ Custom interface.

**User Option** *emacspeak-bookshare-user-id* [Variable]  
Bookshare user Id.

**User Option** *emacspeak-url-template-google-calendar-uri* [Variable]  
URI for accessing mobile version of Google Calendar. Google Apps users should set this to `http://www.google.com/calendar/a/<my-corp>/m?output=xhtml`

**User Option** *emacspeak-url-template-reading-list-opml* [Variable]  
OPML feed location to use for our Custom Search.

**User Option** *emacspeak-url-template-weather-city-state* [Variable]  
Default city/state for weather forecasts

## 12.52 emacspeak-vm Options

Automatically generated documentation for options defined in module emacspeak-vm. These options are customizable via Emacs’ Custom interface.

**User Option** *emacspeak-vm-cal2text* [Variable]  
Executable that converts calendar invitations on standard input to plain text.

**User Option** *emacspeak-vm-customize-mime-settings* [Variable]  
Non-nil will cause Emacspeak to configure VM mime settings to match what the author of Emacspeak uses.

**User Option** *emacspeak-vm-doc2text* [Variable]  
Executable that converts MSWord documents on standard input to plain text using `wvText`.

**User Option** *emacspeak-vm-headers-strip-octals* [Variable]  
Specify whether non-ascii chars should be stripped when speaking email headers.

**User Option** *emacspeak-vm-pdf2text* [Variable]  
Executable that converts PDF on standard input to plain text using `pdftotext`.

**User Option** *emacspeak-vm-ppt2html* [Variable]  
Executable that converts MSPPT documents on standard input to HTML using `xlhtml`.

**User Option** *emacspeak-vm-use-raman-settings* [Variable]  
Should VM use the customizations used by the author of Emacspeak.

**User Option** *emacspeak-vm-voice-lock-messages* [Variable]  
 Set this to T if you want messages automatically voice locked. Note that some badly formed mime messages cause trouble.

**User Option** *emacspeak-vm-xls2html* [Variable]  
 Executable that converts MSXL documents on standard input to HTML using xlshtml.

### 12.53 emacspeak-w3 Options

Commentary: Ensure that speech support for W3 gets installed and loaded correctly. The emacs W3 browser comes with builtin support for Emacspeak and ACSS

Automatically generated documentation for options defined in module emacspeak-w3. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-w3-base-uri-pronunciation* [Variable]  
 Custom pronunciation for base URIs in w3 buffers.

**User Option** *emacspeak-w3-create-imenu-index* [Variable]  
 Create IMenu index by default.

**User Option** *emacspeak-w3-lwp-request* [Variable]  
 LWP Request command from perl LWP.

**User Option** *emacspeak-w3-lynx-program* [Variable]  
 Name of lynx executable

**User Option** *emacspeak-w3-punctuation-mode* [Variable]  
 Pronunciation mode to use for W3 buffers.

**User Option** *emacspeak-w3-tidy-html* [Variable]  
 Tidy HTML before rendering.

**User Option** *emacspeak-w3-tidy-options* [Variable]  
 Options to pass to tidy program

**User Option** *emacspeak-w3-tidy-program* [Variable]  
 Name of tidy executable

### 12.54 emacspeak-w3m Options

Commentary:

Automatically generated documentation for options defined in module emacspeak-w3m. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-w3m-speak-titles-on-switch* [Variable]  
 Speak the document title when switching between w3m buffers. If non-nil, switching between w3m buffers will speak the title instead of the modeline.

## 12.55 emacspeak-we Options

Commentary: we is for webedit Invoke XSLT to edit/transform Web pages before they get rendered. we makes emacspeak's webedit layer independent of a given Emacs web browser like W3 or W3M This module will use the abstraction provided by browse-url to handle Web pages. Module emacspeak-webutils provides the needed additional abstractions not already covered by browse-url

Automatically generated documentation for options defined in module emacspeak-we. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-we-cleanup-bogus-quotes* [Variable]  
Clean up bogus Unicode chars for magic quotes.

**User Option** *emacspeak-we-media-stream-suffixes* [Variable]  
Suffixes that identify URLs to media streams.

**User Option** *emacspeak-we-recent-xpath-filter* [Variable]  
Caches most recently used xpath filter. Can be customized to set up initial default.

**User Option** *emacspeak-we-xsl-p* [Variable]  
T means we apply XSL before displaying HTML.

**User Option** *emacspeak-we-xsl-transform* [Variable]  
Specifies transform to use before displaying a page. Nil means no transform is used.

## 12.56 emacspeak-websearch Options

Commentary: This module provides utility functions for searching the WWW

Automatically generated documentation for options defined in module emacspeak-websearch. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-websearch-emapspeak-my-location* [Variable]  
Specifies location near we look by default.

**User Option** *emacspeak-websearch-google-number-of-results* [Variable]  
Number of results to return from google search.

**User Option** *emacspeak-websearch-google-options* [Variable]  
Additional options to pass to Google e.g. &xx=yy...

**User Option** *emacspeak-websearch-personal-portfolio* [Variable]  
Set this to the stock tickers you want to check by default.

## 12.57 emacspeak-webutils Options

Commentary: This module provides common Web utilities for emacspeak. This is to avoid duplication of code between emacspeak-w3.el and emacspeak-w3m.el

Automatically generated documentation for options defined in module emacspeak-webutils. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-atom-feeds* [Variable]  
Table of ATOM feeds.

- User Option** *emacspeak-atom-view-xsl* [Variable]  
XSL stylesheet used for viewing Atom Feeds.
- User Option** *emacspeak-rss-feeds* [Variable]  
Table of RSS feeds.
- User Option** *emacspeak-rss-unescape-html* [Variable]  
Fix malformed XML that results from sites attempting to unescape HTML tags.
- User Option** *emacspeak-webutils-charent-alist* [Variable]  
Entities to unescape when treating badly escaped XML.

## 12.58 emacspeak-wizards Options

Commentary: Contains various wizards for the Emacspeak desktop.

Automatically generated documentation for options defined in module emacspeak-wizards. These options are customizable via Emacs' Custom interface.

- User Option** *emacspeak-clipboard-file* [Variable]  
File used to save Emacspeak clipboard. The emacspeak clipboard provides a convenient mechanism for exchanging information between different Emacs sessions.
- User Option** *emacspeak-curl-cookie-store* [Variable]  
Cookie store used by Curl.
- User Option** *emacspeak-cvs-local-directory* [Variable]  
Directory where we download the snapshot.
- User Option** *emacspeak-cvs-local-directory-pattern* [Variable]  
Pattern from which name of local download directory is build. %s is replaced by the project name.
- User Option** *emacspeak-emergency-tts-server* [Variable]  
TTS server to use in an emergency. Set this to a TTS server that is known to work at all times. If you are debugging another speech server and that server gets wedged for some reason, you can use command emacspeak-emergency-tts-restart to get speech back using the reliable TTS server. It's useful to bind the above command to a convenient key.
- User Option** *emacspeak-speak-show-active-network-interfaces-addresses* [Variable]  
Command that displays address of a specific interface.
- User Option** *emacspeak-speak-show-active-network-interfaces-command* [Variable]  
Command that displays names of active network interfaces.
- User Option** *emacspeak-speak-telephone-directory* [Variable]  
File holding telephone directory. This is just a text file, and we use grep to search it.
- User Option** *emacspeak-speak-telephone-directory-command* [Variable]  
Command used to look up names in the telephone directory.

- User Option** *emacspeak-ssh-tts-server* [Variable]  
SSH TTS server to use by default.
- User Option** *emacspeak-wizards-curl-program* [Variable]  
Name of curl executable.
- User Option** *emacspeak-wizards-dvi2txt-program* [Variable]  
Program for converting dvi to txt. Set this to nil if you do not want to use the DVI wizard.
- User Option** *emacspeak-wizards-find-switches-that-need-quoting* [Variable]  
Find switches whose args need quoting.
- User Option** *emacspeak-wizards-find-switches-widget* [Variable]  
Widget to get find switch.
- User Option** *emacspeak-wizards-links-program* [Variable]  
Name of links executable.
- User Option** *emacspeak-wizards-linux-howto-directory* [Variable]  
Root of Linux Howtos.
- User Option** *emacspeak-wizards-lynx-program* [Variable]  
Lynx executable.
- User Option** *emacspeak-wizards-pdf-to-text-options* [Variable]  
options to Command for running pdftotext.
- User Option** *emacspeak-wizards-pdf-to-text-program* [Variable]  
Command for running pdftotext.
- User Option** *emacspeak-wizards-personal-portfolio* [Variable]  
Set this to the stock tickers you want to check by default.
- User Option** *emacspeak-wizards-ppthtml-program* [Variable]  
Program for converting PPT to HTML. Set this to nil if you do not want to use the PPTHTML wizard.
- User Option** *emacspeak-wizards-quote-command* [Variable]  
Command for pulling up detailed stock quotes. this requires Perl module Finance::YahooQuote.
- User Option** *emacspeak-wizards-quote-row-filter* [Variable]  
Format used to filter rows.
- User Option** *emacspeak-wizards-spot-words-extension* [Variable]  
Default file extension used when spotting words.
- User Option** *emacspeak-wizards-tramp-locations* [Variable]  
Tramp locations used by Emacspeak tramp wizard. Locations added here via custom can be opened using command *emacspeak-wizards-tramp-open-location* bound to M-x *emacspeak-wizards-tramp-open-location*.

- User Option** *emacspeak-wizards-vc-viewer-command* [Variable]  
Command line for dumping out virtual console.
- User Option** *emacspeak-wizards-vpn-end-command* [Variable]  
Command that brings down a VPN connection.
- User Option** *emacspeak-wizards-vpn-start-command* [Variable]  
Command that brings up a VPN connection.
- User Option** *emacspeak-wizards-xlhtml-program* [Variable]  
Program for converting XL to HTML. Set this to nil if you do not want to use the XLHTML wizard.

## 12.59 emacspeak-xml-shell Options

Automatically generated documentation for options defined in module emacspeak-xml-shell. These options are customizable via Emacs' Custom interface.

- User Option** *emacspeak-xml-shell-command* [Variable]  
Executable that provides the XML browser shell. Default is xmllint. If you want an XML Shell on steroids get XSH and use emacs custom to customize the default to be xsh.
- User Option** *emacspeak-xml-shell-hooks* [Variable]  
Start up hooks run after XML browser process is started.
- User Option** *emacspeak-xml-shell-options* [Variable]  
Command-line options for XML browse command.
- User Option** *emacspeak-xml-shell-xslt* [Variable]  
XSL transform to apply when displaying current node.

## 12.60 emacspeak-xslt Options

Automatically generated documentation for options defined in module emacspeak-xslt. These options are customizable via Emacs' Custom interface.

- User Option** *emacspeak-xslt-keep-errors* [Variable]  
If non-nil, xslt errors will be preserved in an errors buffer.
- User Option** *emacspeak-xslt-nuke-null-char* [Variable]  
If T null chars in the region will be nuked. This is useful when handling bad HTML.
- User Option** *emacspeak-xslt-options* [Variable]  
Options passed to xsltproc.
- User Option** *emacspeak-xslt-program* [Variable]  
Name of XSLT transformation engine.
- User Option** *emacspeak-xslt-use-wget-to-download* [Variable]  
Set to T if you want to avoid URL downloader bugs in libxml2. There is a bug that bites when using Yahoo Maps that wget can work around.

## 12.61 emacspeak Options

Commentary: The complete audio desktop.

Emacspeak extends Emacs to be a fully functional audio desktop. This is the main emacspeak module. It actually does very little: It loads the various parts of the system.

Automatically generated documentation for options defined in module emacspeak. These options are customizable via Emacs' Custom interface.

**User Option** *emacspeak-media-player* [Variable]

Default media player to use. This is a Lisp function that takes a resource locator.

**User Option** *emacspeak-play-emacspeak-startup-icon* [Variable]

If set to T, emacspeak plays its icon as it launches.

**User Option** *emacspeak-startup-hook* [Variable]

Hook to run after starting emacspeak.

## 12.62 voice-setup Options

Commentary: A voice is to audio as a font is to a visual display. A personality is to audio as a face is to a visual display.

Voice-lock-mode is a minor mode that causes your comments to be spoken in one personality, strings in another, reserved words in another, documentation strings in another, and so on.

Comments will be spoken in 'emacspeak-voice-lock-comment-personality'. Strings will be spoken in 'emacspeak-voice-lock-string-personality'. Doc strings will be spoken in 'emacspeak-voice-lock-doc-string-personality'. Function and variable names (in their defining forms) will be spoken in 'emacspeak-voice-lock-function-name-personality'. Reserved words will be spoken in 'emacspeak-voice-lock-keyword-personality'.

To make the text you type be voiceified, use M-x voice-lock-mode. When this minor mode is on, the voices of the current line are updated with every insertion or deletion.

How faces map to voices: TTS engine specific modules e.g., dectalk-voices.el and outloud-voices.el define a standard set of voice names. This module maps standard "personality" names to these pre-defined voices. It does this via special form def-voice-font which takes a personality name, a voice name and a face name to set up the mapping between face and personality, and personality and voice. Newer Emacspeak modules should use voice-setup-add-map when defining face->personality mappings. Older code calls def-voice-font directly, but over time those calls will be changed to the more succinct form provided by voice-setup-add-map. For use from other modules, also see function voice-setup-map-face which is useful when mapping a single face. Both voice-setup-add-map and voice-setup-map-face call special form def-voice-font.

Special form def-voice-font sets up the personality name to be available via custom. new voices can be defined using CSS style specifications see special form defvoice Voices defined via defvoice can be customized via custom see the documentation for defvoice.

Automatically generated documentation for options defined in module voice-setup. These options are customizable via Emacs' Custom interface.

|                                                                     |            |
|---------------------------------------------------------------------|------------|
| <b>User Option</b> <i>voice-animate-extra-settings</i>              | [Variable] |
| Adds extra animation current voice.                                 |            |
| <b>User Option</b> <i>voice-animate-medium-settings</i>             | [Variable] |
| Adds medium animation current voice.                                |            |
| <b>User Option</b> <i>voice-animate-settings</i>                    | [Variable] |
| Animates current voice.                                             |            |
| <b>User Option</b> <i>voice-annotate-settings</i>                   | [Variable] |
| Indicate annotation.                                                |            |
| <b>User Option</b> <i>voice-bolden-and-animate-settings</i>         | [Variable] |
| Bolden and animate current voice.                                   |            |
| <b>User Option</b> <i>voice-bolden-extra-settings</i>               | [Variable] |
| Extra bolden current voice.                                         |            |
| <b>User Option</b> <i>voice-bolden-medium-settings</i>              | [Variable] |
| Add medium bolden current voice.                                    |            |
| <b>User Option</b> <i>voice-bolden-settings</i>                     | [Variable] |
| Bolden current voice.                                               |            |
| <b>User Option</b> <i>voice-brighten-extra-settings</i>             | [Variable] |
| Extra brighten current voice.                                       |            |
| <b>User Option</b> <i>voice-brighten-medium-settings</i>            | [Variable] |
| Brighten current voice.                                             |            |
| <b>User Option</b> <i>voice-brighten-settings</i>                   | [Variable] |
| Brighten current voice.                                             |            |
| <b>User Option</b> <i>voice-indent-settings</i>                     | [Variable] |
| Indicate indentation .                                              |            |
| <b>User Option</b> <i>voice-lighten-extra-settings</i>              | [Variable] |
| Add extra lighten current voice.                                    |            |
| <b>User Option</b> <i>voice-lighten-medium-settings</i>             | [Variable] |
| Add medium lighten current voice.                                   |            |
| <b>User Option</b> <i>voice-lighten-settings</i>                    | [Variable] |
| Lighten current voice.                                              |            |
| <b>User Option</b> <i>voice-lock-mode</i>                           | [Variable] |
| Determines if property personality results in text being voicified. |            |
| <b>User Option</b> <i>voice-lock-overlay-0-settings</i>             | [Variable] |
| Overlay voice that sets dimension 0 of ACSS structure to 8.         |            |
| <b>User Option</b> <i>voice-lock-overlay-1-settings</i>             | [Variable] |
| Overlay voice that sets dimension 1 of ACSS structure to 8.         |            |

|                                                                  |            |
|------------------------------------------------------------------|------------|
| <b>User Option</b> <i>voice-lock-overlay-2-settings</i>          | [Variable] |
| Overlay voice that sets dimension 2 of ACSS structure to 8.      |            |
| <b>User Option</b> <i>voice-lock-overlay-3-settings</i>          | [Variable] |
| Overlay voice that sets dimension 3 of ACSS structure to 8.      |            |
| <b>User Option</b> <i>voice-monotone-medium-settings</i>         | [Variable] |
| Turns current voice into a medium monotone.                      |            |
| <b>User Option</b> <i>voice-monotone-settings</i>                | [Variable] |
| Turns current voice into a monotone and speaks all punctuations. |            |
| <b>User Option</b> <i>voice-punctuations-all-settings</i>        | [Variable] |
| Turns current voice into one that speaks all punctuations.       |            |
| <b>User Option</b> <i>voice-punctuations-none-settings</i>       | [Variable] |
| Turns current voice into one that speaks no punctuations.        |            |
| <b>User Option</b> <i>voice-punctuations-some-settings</i>       | [Variable] |
| Turns current voice into one that speaks some punctuations.      |            |
| <b>User Option</b> <i>voice-setup-sample-text</i>                | [Variable] |
| Sample text used when displaying available voices.               |            |
| <b>User Option</b> <i>voice-smoothen-extra-settings</i>          | [Variable] |
| Extra smoothen current voice.                                    |            |
| <b>User Option</b> <i>voice-smoothen-medium-settings</i>         | [Variable] |
| Add medium smoothen current voice.                               |            |
| <b>User Option</b> <i>voice-smoothen-settings</i>                | [Variable] |
| Smoothen current voice.                                          |            |
| <b>User Option</b> <i>voice-womanize-1-settings</i>              | [Variable] |
| Apply first female voice.                                        |            |

Automatically generated documentation for options defined in module nil. These options are customizable via Emacs' Custom interface.

|                                                                                                                                                        |            |
|--------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>User Option</b> <i>emacspeak-Buffer-menu-buffer-voice</i>                                                                                           | [Variable] |
| Personality used for Buffer-menu-buffer This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.  |            |
| <b>User Option</b> <i>emacspeak-bbdb-company-voice</i>                                                                                                 | [Variable] |
| Personality used for bbdb-company This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.    |            |
| <b>User Option</b> <i>emacspeak-bbdb-field-name-voice</i>                                                                                              | [Variable] |
| Personality used for bbdb-field-name This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings. |            |

- User Option** *emacspeak-bbdb-field-value-voice* [Variable]  
 Personality used for `bbdb-field-value` This personality uses `voice-animate` whose effect can be changed globally by customizing `voice-animate-settings`.
- User Option** *emacspeak-bbdb-name-voice* [Variable]  
 Personality used for `bbdb-name` This personality uses `voice-bolden` whose effect can be changed globally by customizing `voice-bolden-settings`.
- User Option** *emacspeak-bold-italic-voice* [Variable]  
 Personality used for `bold-italic` This personality uses `voice-bolden-and-animate` whose effect can be changed globally by customizing `voice-bolden-and-animate-settings`.
- User Option** *emacspeak-bold-voice* [Variable]  
 Personality used for `bold` This personality uses `voice-bolden` whose effect can be changed globally by customizing `voice-bolden-settings`.
- User Option** *emacspeak-button-voice* [Variable]  
 Personality used for `button` This personality uses `voice-bolden` whose effect can be changed globally by customizing `voice-bolden-settings`.
- User Option** *emacspeak-c-nonbreakable-space-personality* [Variable]  
 Personality used for `c-nonbreakable-space-face` This personality uses `voice-brighten` whose effect can be changed globally by customizing `voice-brighten-settings`.
- User Option** *emacspeak-calendar-today-voice* [Variable]  
 Personality used for `calendar-today` This personality uses `voice-lighten` whose effect can be changed globally by customizing `voice-lighten-settings`.
- User Option** *emacspeak-comint-highlight-input-voice* [Variable]  
 Personality used for `comint-highlight-input` This personality uses `voice-bolden-medium` whose effect can be changed globally by customizing `voice-bolden-medium-settings`.
- User Option** *emacspeak-comint-highlight-prompt-voice* [Variable]  
 Personality used for `comint-highlight-prompt` This personality uses `voice-monotone-medium` whose effect can be changed globally by customizing `voice-monotone-medium-settings`.
- User Option** *emacspeak-custom-button-mouse-voice* [Variable]  
 Personality used for `custom-button-mouse` This personality uses `voice-bolden-medium` whose effect can be changed globally by customizing `voice-bolden-medium-settings`.
- User Option** *emacspeak-custom-button-pressed-unraised-voice* [Variable]  
 Personality used for `custom-button-pressed-unraised` This personality uses `voice-bolden-extra` whose effect can be changed globally by customizing `voice-bolden-extra-settings`.
- User Option** *emacspeak-custom-button-pressed-voice* [Variable]  
 Personality used for `custom-button-pressed` This personality uses `voice-bolden-extra` whose effect can be changed globally by customizing `voice-bolden-extra-settings`.

- User Option** *emacspeak-custom-button-unraised-voice* [Variable]  
Personality used for custom-button-unraised This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-custom-button-voice* [Variable]  
Personality used for custom-button This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-custom-changed-voice* [Variable]  
Personality used for custom-changed This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-custom-comment-personality* [Variable]  
Personality used for custom-comment-face This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings.
- User Option** *emacspeak-custom-comment-tag-personality* [Variable]  
Personality used for custom-comment-tag-face This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.
- User Option** *emacspeak-custom-comment-tag-voice* [Variable]  
Personality used for custom-comment-tag This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.
- User Option** *emacspeak-custom-documentation-voice* [Variable]  
Personality used for custom-documentation This personality uses voice-brighten-medium whose effect can be changed globally by customizing voice-brighten-medium-settings.
- User Option** *emacspeak-custom-face-tag-voice* [Variable]  
Personality used for custom-face-tag This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-custom-group-tag-1-voice* [Variable]  
Personality used for custom-group-tag-1 This personality uses voice-lighten-medium whose effect can be changed globally by customizing voice-lighten-medium-settings.
- User Option** *emacspeak-custom-group-tag-personality* [Variable]  
Personality used for custom-group-tag-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-custom-group-tag-voice* [Variable]  
Personality used for custom-group-tag This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-custom-invalid-voice* [Variable]  
Personality used for custom-invalid This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.

- User Option** *emacspeak-custom-link-voice* [Variable]  
 Personality used for custom-link This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-custom-modified-personality* [Variable]  
 Personality used for custom-modified-face This personality uses voice-lighten-medium whose effect can be changed globally by customizing voice-lighten-medium-settings.
- User Option** *emacspeak-custom-modified-voice* [Variable]  
 Personality used for custom-modified This personality uses voice-lighten-medium whose effect can be changed globally by customizing voice-lighten-medium-settings.
- User Option** *emacspeak-custom-rogue-voice* [Variable]  
 Personality used for custom-rogue This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.
- User Option** *emacspeak-custom-saved-voice* [Variable]  
 Personality used for custom-saved This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-custom-set-voice* [Variable]  
 Personality used for custom-set This personality uses voice-smoothen-medium whose effect can be changed globally by customizing voice-smoothen-medium-settings.
- User Option** *emacspeak-custom-state-voice* [Variable]  
 Personality used for custom-state This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-custom-variable-button-voice* [Variable]  
 Personality used for custom-variable-button This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-custom-variable-tag-voice* [Variable]  
 Personality used for custom-variable-tag This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-diary-personality* [Variable]  
 Personality used for diary-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-dired-directory-voice* [Variable]  
 Personality used for dired-directory This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-dired-flagged-voice* [Variable]  
 Personality used for dired-flagged This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.

- User Option** *emacspeak-dired-header-voice* [Variable]  
 Personality used for dired-header This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-dired-ignored-voice* [Variable]  
 Personality used for dired-ignored This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-dired-mark-voice* [Variable]  
 Personality used for dired-mark This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-dired-marked-voice* [Variable]  
 Personality used for dired-marked This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.
- User Option** *emacspeak-dired-symlink-voice* [Variable]  
 Personality used for dired-symlink This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.
- User Option** *emacspeak-dired-warning-voice* [Variable]  
 Personality used for dired-warning This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.
- User Option** *emacspeak-eperiodic-discovered-after-personality* [Variable]  
 Personality used for eperiodic-discovered-after-face This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-eperiodic-discovered-before-personality* [Variable]  
 Personality used for eperiodic-discovered-before-face This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-eperiodic-discovered-in-personality* [Variable]  
 Personality used for eperiodic-discovered-in-face This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-eperiodic-f-block-personality* [Variable]  
 Personality used for eperiodic-f-block-face This personality uses voice-lighten-medium whose effect can be changed globally by customizing voice-lighten-medium-settings.
- User Option** *emacspeak-eperiodic-gas-personality* [Variable]  
 Personality used for eperiodic-gas-face This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-eperiodic-group-number-personality* [Variable]  
 Personality used for eperiodic-group-number-face This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.

- User Option** *emacspeak-eperiodic-header-personality* [Variable]  
 Personality used for eperiodic-header-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-eperiodic-liquid-personality* [Variable]  
 Personality used for eperiodic-liquid-face This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-eperiodic-p-block-personality* [Variable]  
 Personality used for eperiodic-p-block-face This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.
- User Option** *emacspeak-eperiodic-period-number-personality* [Variable]  
 Personality used for eperiodic-period-number-face This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-eperiodic-s-block-personality* [Variable]  
 Personality used for eperiodic-s-block-face This personality uses voice-smoothen-medium whose effect can be changed globally by customizing voice-smoothen-medium-settings.
- User Option** *emacspeak-eperiodic-solid-personality* [Variable]  
 Personality used for eperiodic-solid-face This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-eperiodic-unknown-personality* [Variable]  
 Personality used for eperiodic-unknown-face This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.
- User Option** *emacspeak-erc-action-personality* [Variable]  
 Personality used for erc-action-face This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.
- User Option** *emacspeak-erc-bold-personality* [Variable]  
 Personality used for erc-bold-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-erc-dangerous-host-personality* [Variable]  
 Personality used for erc-dangerous-host-face This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.
- User Option** *emacspeak-erc-direct-msg-personality* [Variable]  
 Personality used for erc-direct-msg-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-erc-error-personality* [Variable]  
 Personality used for erc-error-face This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.

- User Option** *emacspeak-erc-input-personality* [Variable]  
Personality used for erc-input-face This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-erc-inverse-personality* [Variable]  
Personality used for erc-inverse-face This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-erc-keyword-personality* [Variable]  
Personality used for erc-keyword-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-erc-notice-personality* [Variable]  
Personality used for erc-notice-face This personality uses (quote inaudible) whose effect can be changed globally by customizing (quote inaudible)-settings.
- User Option** *emacspeak-erc-pal-personality* [Variable]  
Personality used for erc-pal-face This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.
- User Option** *emacspeak-erc-prompt-personality* [Variable]  
Personality used for erc-prompt-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-erc-underline-personality* [Variable]  
Personality used for erc-underline-face This personality uses voice-brighten-medium whose effect can be changed globally by customizing voice-brighten-medium-settings.
- User Option** *emacspeak-eshell-ls-archive-personality* [Variable]  
Personality used for eshell-ls-archive-face This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-eshell-ls-archive-voice* [Variable]  
Personality used for eshell-ls-archive This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-eshell-ls-backup-personality* [Variable]  
Personality used for eshell-ls-backup-face This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings.
- User Option** *emacspeak-eshell-ls-backup-voice* [Variable]  
Personality used for eshell-ls-backup This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings.
- User Option** *emacspeak-eshell-ls-clutter-personality* [Variable]  
Personality used for eshell-ls-clutter-face This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.

- User Option** *emacspeak-eshell-ls-clutter-voice* [Variable]  
Personality used for eshell-ls-clutter This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-eshell-ls-directory-personality* [Variable]  
Personality used for eshell-ls-directory-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-eshell-ls-directory-voice* [Variable]  
Personality used for eshell-ls-directory This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-eshell-ls-executable-personality* [Variable]  
Personality used for eshell-ls-executable-face This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.
- User Option** *emacspeak-eshell-ls-executable-voice* [Variable]  
Personality used for eshell-ls-executable This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.
- User Option** *emacspeak-eshell-ls-missing-personality* [Variable]  
Personality used for eshell-ls-missing-face This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-eshell-ls-missing-voice* [Variable]  
Personality used for eshell-ls-missing This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-eshell-ls-product-personality* [Variable]  
Personality used for eshell-ls-product-face This personality uses voice-lighten-medium whose effect can be changed globally by customizing voice-lighten-medium-settings.
- User Option** *emacspeak-eshell-ls-product-voice* [Variable]  
Personality used for eshell-ls-product This personality uses voice-lighten-medium whose effect can be changed globally by customizing voice-lighten-medium-settings.
- User Option** *emacspeak-eshell-ls-readonly-personality* [Variable]  
Personality used for eshell-ls-readonly-face This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.
- User Option** *emacspeak-eshell-ls-readonly-voice* [Variable]  
Personality used for eshell-ls-readonly This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.
- User Option** *emacspeak-eshell-ls-special-personality* [Variable]  
Personality used for eshell-ls-special-face This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.

- User Option** *emacspeak-eshell-ls-special-voice* [Variable]  
Personality used for eshell-ls-special This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-eshell-ls-symlink-personality* [Variable]  
Personality used for eshell-ls-symlink-face This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-eshell-ls-symlink-voice* [Variable]  
Personality used for eshell-ls-symlink This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-eshell-ls-unreadable-personality* [Variable]  
Personality used for eshell-ls-unreadable-face This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings.
- User Option** *emacspeak-eshell-ls-unreadable-voice* [Variable]  
Personality used for eshell-ls-unreadable This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings.
- User Option** *emacspeak-eshell-prompt-personality* [Variable]  
Personality used for eshell-prompt-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-eshell-prompt-voice* [Variable]  
Personality used for eshell-prompt This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-fixed-pitch-voice* [Variable]  
Personality used for fixed-pitch This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.
- User Option** *emacspeak-fixed-voice* [Variable]  
Personality used for fixed This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.
- User Option** *emacspeak-flyspell-incorrect-personality* [Variable]  
Personality used for flyspell-incorrect-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-cite-1-voice* [Variable]  
Personality used for gnus-cite-1 This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-gnus-cite-10-voice* [Variable]  
Personality used for gnus-cite-10 This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.

- User Option** *emacspeak-gnus-cite-11-voice* [Variable]  
Personality used for gnus-cite-11 This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-gnus-cite-2-voice* [Variable]  
Personality used for gnus-cite-2 This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-gnus-cite-3-voice* [Variable]  
Personality used for gnus-cite-3 This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-gnus-cite-4-voice* [Variable]  
Personality used for gnus-cite-4 This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-cite-5-voice* [Variable]  
Personality used for gnus-cite-5 This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-gnus-cite-6-voice* [Variable]  
Personality used for gnus-cite-6 This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-gnus-cite-7-voice* [Variable]  
Personality used for gnus-cite-7 This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-gnus-cite-8-voice* [Variable]  
Personality used for gnus-cite-8 This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-cite-9-voice* [Variable]  
Personality used for gnus-cite-9 This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-gnus-cite-face-1-voice* [Variable]  
Personality used for gnus-cite-face-1 This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-cite-face-10-voice* [Variable]  
Personality used for gnus-cite-face-10 This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-gnus-cite-face-2-voice* [Variable]  
Personality used for gnus-cite-face-2 This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-gnus-cite-face-3-voice* [Variable]  
Personality used for gnus-cite-face-3 This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.

- User Option** *emacspeak-gnus-cite-face-4-voice* [Variable]  
Personality used for gnus-cite-face-4 This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-gnus-cite-face-5-voice* [Variable]  
Personality used for gnus-cite-face-5 This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-cite-face-6-voice* [Variable]  
Personality used for gnus-cite-face-6 This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-gnus-cite-face-7-voice* [Variable]  
Personality used for gnus-cite-face-7 This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-gnus-cite-face-8-voice* [Variable]  
Personality used for gnus-cite-face-8 This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-gnus-cite-face-9-voice* [Variable]  
Personality used for gnus-cite-face-9 This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-emphasis-bold-voice* [Variable]  
Personality used for gnus-emphasis-bold This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.
- User Option** *emacspeak-gnus-emphasis-highlight-words-voice* [Variable]  
Personality used for gnus-emphasis-highlight-words This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-gnus-emphasis-italic-voice* [Variable]  
Personality used for gnus-emphasis-italic This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-gnus-emphasis-strikethru-voice* [Variable]  
Personality used for gnus-emphasis-strikethru This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-emphasis-underline-voice* [Variable]  
Personality used for gnus-emphasis-underline This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.
- User Option** *emacspeak-gnus-group-mail-1-empty-personality* [Variable]  
Personality used for gnus-group-mail-1-empty-face This personality uses default whose effect can be changed globally by customizing default-settings.

- User Option** *emacspeak-gnus-group-mail-1-empty-voice* [Variable]  
Personality used for gnus-group-mail-1-empty This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-mail-1-personality* [Variable]  
Personality used for gnus-group-mail-1-face This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-group-mail-1-voice* [Variable]  
Personality used for gnus-group-mail-1 This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-group-mail-2-empty-personality* [Variable]  
Personality used for gnus-group-mail-2-empty-face This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-gnus-group-mail-2-empty-voice* [Variable]  
Personality used for gnus-group-mail-2-empty This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-mail-2-personality* [Variable]  
Personality used for gnus-group-mail-2-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-mail-2-voice* [Variable]  
Personality used for gnus-group-mail-2 This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-group-mail-3-empty-personality* [Variable]  
Personality used for gnus-group-mail-3-empty-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-mail-3-empty-voice* [Variable]  
Personality used for gnus-group-mail-3-empty This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-mail-3-personality* [Variable]  
Personality used for gnus-group-mail-3-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-mail-3-voice* [Variable]  
Personality used for gnus-group-mail-3 This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-group-mail-low-empty-personality* [Variable]  
Personality used for gnus-group-mail-low-empty-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

- User Option** *emacspeak-gnus-group-mail-low-empty-voice* [Variable]  
Personality used for gnus-group-mail-low-empty This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-mail-low-personality* [Variable]  
Personality used for gnus-group-mail-low-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-mail-low-voice* [Variable]  
Personality used for gnus-group-mail-low This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-group-news-1-empty-personality* [Variable]  
Personality used for gnus-group-news-1-empty-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-news-1-empty-voice* [Variable]  
Personality used for gnus-group-news-1-empty This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-news-1-personality* [Variable]  
Personality used for gnus-group-news-1-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-news-1-voice* [Variable]  
Personality used for gnus-group-news-1 This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-group-news-2-empty-personality* [Variable]  
Personality used for gnus-group-news-2-empty-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-news-2-empty-voice* [Variable]  
Personality used for gnus-group-news-2-empty This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-news-2-personality* [Variable]  
Personality used for gnus-group-news-2-face This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-news-2-voice* [Variable]  
Personality used for gnus-group-news-2 This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-group-news-3-empty-personality* [Variable]  
Personality used for gnus-group-news-3-empty-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

- User Option** *emacspeak-gnus-group-news-3-empty-voice* [Variable]  
Personality used for gnus-group-news-3-empty This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-news-3-personality* [Variable]  
Personality used for gnus-group-news-3-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-news-3-voice* [Variable]  
Personality used for gnus-group-news-3 This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-group-news-4-empty-personality* [Variable]  
Personality used for gnus-group-news-4-empty-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-news-4-empty-voice* [Variable]  
Personality used for gnus-group-news-4-empty This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-news-4-personality* [Variable]  
Personality used for gnus-group-news-4-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-news-4-voice* [Variable]  
Personality used for gnus-group-news-4 This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-group-news-5-empty-personality* [Variable]  
Personality used for gnus-group-news-5-empty-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-news-5-empty-voice* [Variable]  
Personality used for gnus-group-news-5-empty This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-news-5-personality* [Variable]  
Personality used for gnus-group-news-5-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-group-news-5-voice* [Variable]  
Personality used for gnus-group-news-5 This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-group-news-6-empty-personality* [Variable]  
Personality used for gnus-group-news-6-empty-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

- User Option** *emacspeak-gnus-group-news-6-empty-voice* [Variable]  
 Personality used for gnus-group-news-6-empty This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-news-6-personality* [Variable]  
 Personality used for gnus-group-news-6-face This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-news-6-voice* [Variable]  
 Personality used for gnus-group-news-6 This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-group-news-low-empty-personality* [Variable]  
 Personality used for gnus-group-news-low-empty-face This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-news-low-empty-voice* [Variable]  
 Personality used for gnus-group-news-low-empty This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-news-low-personality* [Variable]  
 Personality used for gnus-group-news-low-face This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-group-news-low-voice* [Variable]  
 Personality used for gnus-group-news-low This personality uses default whose effect can be changed globally by customizing default-settings.
- User Option** *emacspeak-gnus-header-content-personality* [Variable]  
 Personality used for gnus-header-content-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-header-content-voice* [Variable]  
 Personality used for gnus-header-content This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-header-from-personality* [Variable]  
 Personality used for gnus-header-from-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-header-from-voice* [Variable]  
 Personality used for gnus-header-from This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-header-name-personality* [Variable]  
 Personality used for gnus-header-name-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.

- User Option** *emacspeak-gnus-header-name-voice* [Variable]  
Personality used for gnus-header-name This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-gnus-header-newsgroups-personality* [Variable]  
Personality used for gnus-header-newsgroups-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-header-newsgroups-voice* [Variable]  
Personality used for gnus-header-newsgroups This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-header-subject-personality* [Variable]  
Personality used for gnus-header-subject-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-header-subject-voice* [Variable]  
Personality used for gnus-header-subject This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-server-agent-personality* [Variable]  
Personality used for gnus-server-agent-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-server-agent-voice* [Variable]  
Personality used for gnus-server-agent This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-server-closed-personality* [Variable]  
Personality used for gnus-server-closed-face This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-gnus-server-closed-voice* [Variable]  
Personality used for gnus-server-closed This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-gnus-server-denied-personality* [Variable]  
Personality used for gnus-server-denied-face This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-server-denied-voice* [Variable]  
Personality used for gnus-server-denied This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-server-offline-personality* [Variable]  
Personality used for gnus-server-offline-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.

- User Option** *emacspeak-gnus-server-offline-voice* [Variable]  
Personality used for gnus-server-offline This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-gnus-server-opened-personality* [Variable]  
Personality used for gnus-server-opened-face This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-gnus-server-opened-voice* [Variable]  
Personality used for gnus-server-opened This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-gnus-signature-personality* [Variable]  
Personality used for gnus-signature-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-gnus-signature-voice* [Variable]  
Personality used for gnus-signature This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-gnus-summary-cancelled-personality* [Variable]  
Personality used for gnus-summary-cancelled-face This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-summary-cancelled-voice* [Variable]  
Personality used for gnus-summary-cancelled This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-summary-high-ancient-personality* [Variable]  
Personality used for gnus-summary-high-ancient-face This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-gnus-summary-high-ancient-voice* [Variable]  
Personality used for gnus-summary-high-ancient This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-gnus-summary-high-read-personality* [Variable]  
Personality used for gnus-summary-high-read-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-summary-high-read-voice* [Variable]  
Personality used for gnus-summary-high-read This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-summary-high-ticked-personality* [Variable]  
Personality used for gnus-summary-high-ticked-face This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.

- User Option** *emacspeak-gnus-summary-high-ticked-voice* [Variable]  
 Personality used for gnus-summary-high-ticked This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.
- User Option** *emacspeak-gnus-summary-high-undownloaded-personality* [Variable]  
 Personality used for gnus-summary-high-undownloaded-face This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.
- User Option** *emacspeak-gnus-summary-high-undownloadedvoice-bolden-and-animate-voice* [Variable]  
 Personality used for gnus-summary-high-undownloadedvoice-bolden-and-animate This personality uses nil whose effect can be changed globally by customizing nil-settings.
- User Option** *emacspeak-gnus-summary-high-unread-personality* [Variable]  
 Personality used for gnus-summary-high-unread-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-summary-high-unread-voice* [Variable]  
 Personality used for gnus-summary-high-unread This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.
- User Option** *emacspeak-gnus-summary-low-ancient-personality* [Variable]  
 Personality used for gnus-summary-low-ancient-face This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-gnus-summary-low-ancient-voice* [Variable]  
 Personality used for gnus-summary-low-ancient This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-gnus-summary-low-read-personality* [Variable]  
 Personality used for gnus-summary-low-read-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-summary-low-read-voice* [Variable]  
 Personality used for gnus-summary-low-read This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-summary-low-ticked-personality* [Variable]  
 Personality used for gnus-summary-low-ticked-face This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-gnus-summary-low-ticked-voice* [Variable]  
 Personality used for gnus-summary-low-ticked This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.

- User Option** *emacspeak-gnus-summary-low-undownloaded-personality* [Variable]  
Personality used for gnus-summary-low-undownloaded-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-summary-low-undownloaded-voice* [Variable]  
Personality used for gnus-summary-low-undownloaded This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.
- User Option** *emacspeak-gnus-summary-low-unread-personality* [Variable]  
Personality used for gnus-summary-low-unread-face This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-gnus-summary-low-unread-voice* [Variable]  
Personality used for gnus-summary-low-unread This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-gnus-summary-normal-ancient-personality* [Variable]  
Personality used for gnus-summary-normal-ancient-face This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-gnus-summary-normal-ancient-voice* [Variable]  
Personality used for gnus-summary-normal-ancient This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-gnus-summary-normal-read-personality* [Variable]  
Personality used for gnus-summary-normal-read-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-summary-normal-read-voice* [Variable]  
Personality used for gnus-summary-normal-read This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-gnus-summary-normal-ticked-personality* [Variable]  
Personality used for gnus-summary-normal-ticked-face This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-gnus-summary-normal-ticked-voice* [Variable]  
Personality used for gnus-summary-normal-ticked This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.
- User Option** *emacspeak-gnus-summary-normal-undownloaded-personality* [Variable]  
Personality used for gnus-summary-normal-undownloaded-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

- User Option** *emacspeak-gnus-summary-normal-undownloaded-voice* [Variable]  
 Personality used for gnus-summary-normal-undownloaded This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.
- User Option** *emacspeak-gnus-summary-selected-personality* [Variable]  
 Personality used for gnus-summary-selected-face This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.
- User Option** *emacspeak-gnus-summary-selected-voice* [Variable]  
 Personality used for gnus-summary-selected This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.
- User Option** *emacspeak-help-argument-name-voice* [Variable]  
 Personality used for help-argument-name This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-highlight-voice* [Variable]  
 Personality used for highlight This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-holiday-personality* [Variable]  
 Personality used for holiday-face This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.
- User Option** *emacspeak-info-header-node-voice* [Variable]  
 Personality used for info-header-node This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-info-header-xref-voice* [Variable]  
 Personality used for info-header-xref This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-info-menu-5-voice* [Variable]  
 Personality used for info-menu-5 This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-info-menu-header-voice* [Variable]  
 Personality used for info-menu-header This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-info-menu-star-voice* [Variable]  
 Personality used for info-menu-star This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-info-node-voice* [Variable]  
 Personality used for info-node This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.

- User Option** *emacspeak-info-title-1-voice* [Variable]  
 Personality used for info-title-1 This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-info-title-2-voice* [Variable]  
 Personality used for info-title-2 This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-info-title-3-voice* [Variable]  
 Personality used for info-title-3 This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-info-title-4-voice* [Variable]  
 Personality used for info-title-4 This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-info-xref-visited-voice* [Variable]  
 Personality used for info-xref-visited This personality uses voice-animate-medium whose effect can be changed globally by customizing voice-animate-medium-settings.
- User Option** *emacspeak-info-xref-voice* [Variable]  
 Personality used for info-xref This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.
- User Option** *emacspeak-ispell-highlight-personality* [Variable]  
 Personality used for ispell-highlight-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-italic-voice* [Variable]  
 Personality used for italic This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-match-voice* [Variable]  
 Personality used for match This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-message-cited-text-personality* [Variable]  
 Personality used for message-cited-text-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-cited-text-voice* [Variable]  
 Personality used for message-cited-text This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-header-cc-personality* [Variable]  
 Personality used for message-header-cc-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-header-cc-voice* [Variable]  
 Personality used for message-header-cc This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.

- User Option** *emacspeak-message-header-name-personality* [Variable]  
 Personality used for message-header-name-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-message-header-name-voice* [Variable]  
 Personality used for message-header-name This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-message-header-newsgroups-personality* [Variable]  
 Personality used for message-header-newsgroups-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-header-newsgroups-voice* [Variable]  
 Personality used for message-header-newsgroups This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-header-other-personality* [Variable]  
 Personality used for message-header-other-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-header-other-voice* [Variable]  
 Personality used for message-header-other This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-header-subject-personality* [Variable]  
 Personality used for message-header-subject-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-header-subject-voice* [Variable]  
 Personality used for message-header-subject This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-header-to-personality* [Variable]  
 Personality used for message-header-to-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-header-to-voice* [Variable]  
 Personality used for message-header-to This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-header-xheader-personality* [Variable]  
 Personality used for message-header-xheader-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-header-xheader-voice* [Variable]  
 Personality used for message-header-xheader This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-message-mml-personality* [Variable]  
 Personality used for message-mml-face This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.

- User Option** *emacspeak-message-mml-voice* [Variable]  
Personality used for message-mml This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-message-separator-personality* [Variable]  
Personality used for message-separator-face This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-message-separator-voice* [Variable]  
Personality used for message-separator This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-outline-1-voice* [Variable]  
Personality used for outline-1 This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-outline-2-voice* [Variable]  
Personality used for outline-2 This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-outline-3-voice* [Variable]  
Personality used for outline-3 This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-outline-4-voice* [Variable]  
Personality used for outline-4 This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-outline-5-voice* [Variable]  
Personality used for outline-5 This personality uses voice-lighten-medium whose effect can be changed globally by customizing voice-lighten-medium-settings.
- User Option** *emacspeak-outline-6-voice* [Variable]  
Personality used for outline-6 This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-region-voice* [Variable]  
Personality used for region This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-shadow-voice* [Variable]  
Personality used for shadow This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings.
- User Option** *emacspeak-term-underline-voice* [Variable]  
Personality used for term-underline This personality uses voice-brighten-medium whose effect can be changed globally by customizing voice-brighten-medium-settings.
- User Option** *emacspeak-tnt-buddy-list-active-personality* [Variable]  
Personality used for tnt-buddy-list-active-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.

- User Option** *emacspeak-tnt-buddy-list-away-personality* [Variable]  
 Personality used for tnt-buddy-list-away-face This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.
- User Option** *emacspeak-tnt-buddy-list-idle-personality* [Variable]  
 Personality used for tnt-buddy-list-idle-face This personality uses voice-smoothen-medium whose effect can be changed globally by customizing voice-smoothen-medium-settings.
- User Option** *emacspeak-tnt-buddy-list-inactive-personality* [Variable]  
 Personality used for tnt-buddy-list-inactive-face This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings.
- User Option** *emacspeak-tnt-buddy-list-pounce-personality* [Variable]  
 Personality used for tnt-buddy-list-pounce-face This personality uses (quote ursula) whose effect can be changed globally by customizing (quote ursula)-settings.
- User Option** *emacspeak-tnt-my-name-personality* [Variable]  
 Personality used for tnt-my-name-face This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-underline-voice* [Variable]  
 Personality used for underline This personality uses voice-lighten-medium whose effect can be changed globally by customizing voice-lighten-medium-settings.
- User Option** *emacspeak-voice-latex-bold-personality* [Variable]  
 Personality used for font-latex-bold-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-voice-latex-doctex-documentation-personality* [Variable]  
 Personality used for font-latex-doctex-documentation-face This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings.
- User Option** *emacspeak-voice-latex-doctex-preprocessor-personality* [Variable]  
 Personality used for font-latex-doctex-preprocessor-face This personality uses voice-brighten-medium whose effect can be changed globally by customizing voice-brighten-medium-settings.
- User Option** *emacspeak-voice-latex-italic-personality* [Variable]  
 Personality used for font-latex-italic-face This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-voice-latex-math-personality* [Variable]  
 Personality used for font-latex-math-face This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.
- User Option** *emacspeak-voice-latex-sedate-personality* [Variable]  
 Personality used for font-latex-sedate-face This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.

- User Option** *emacspeak-voice-latex-string-personality* [Variable]  
 Personality used for font-latex-string-face This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-voice-latex-subscript-personality* [Variable]  
 Personality used for font-latex-subscript-face This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-voice-latex-superscript-personality* [Variable]  
 Personality used for font-latex-superscript-face This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-voice-latex-title-1-personality* [Variable]  
 Personality used for font-latex-title-1-face This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-voice-latex-title-2-personality* [Variable]  
 Personality used for font-latex-title-2-face This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-voice-latex-title-3-personality* [Variable]  
 Personality used for font-latex-title-3-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-voice-latex-title-4-personality* [Variable]  
 Personality used for font-latex-title-4-face This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-voice-latex-verbatim-personality* [Variable]  
 Personality used for font-latex-verbatim-face This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.
- User Option** *emacspeak-voice-latex-warning-personality* [Variable]  
 Personality used for font-latex-warning-face This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.
- User Option** *emacspeak-voice-lock-builtin-personality* [Variable]  
 Personality used for font-lock-builtin-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-voice-lock-comment-delimiter-personality* [Variable]  
 Personality used for font-lock-comment-delimiter-face This personality uses voice-smoothen-medium whose effect can be changed globally by customizing voice-smoothen-medium-settings.
- User Option** *emacspeak-voice-lock-comment-personality* [Variable]  
 Personality used for font-lock-comment-face This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.

- User Option** *emacspeak-voice-lock-constant-personality* [Variable]  
 Personality used for font-lock-constant-face This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-voice-lock-doc-personality* [Variable]  
 Personality used for font-lock-doc-face This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings.
- User Option** *emacspeak-voice-lock-doc-string-personality* [Variable]  
 Personality used for font-lock-doc-string-face This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-voice-lock-function-name-personality* [Variable]  
 Personality used for font-lock-function-name-face This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-voice-lock-keyword-personality* [Variable]  
 Personality used for font-lock-keyword-face This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.
- User Option** *emacspeak-voice-lock-negation-char-personality* [Variable]  
 Personality used for font-lock-negation-char-face This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.
- User Option** *emacspeak-voice-lock-preprocessor-personality* [Variable]  
 Personality used for font-lock-preprocessor-face This personality uses voice-monotone-medium whose effect can be changed globally by customizing voice-monotone-medium-settings.
- User Option** *emacspeak-voice-lock-reference-personality* [Variable]  
 Personality used for font-lock-reference-face This personality uses voice-animate-medium whose effect can be changed globally by customizing voice-animate-medium-settings.
- User Option** *emacspeak-voice-lock-regexp-grouping-backslash* [Variable]  
 Personality used for font-lock-regexp-grouping-backslash This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-voice-lock-regexp-grouping-construct* [Variable]  
 Personality used for font-lock-regexp-grouping-construct This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.

- User Option** *emacspeak-voice-lock-string-personality* [Variable]  
Personality used for font-lock-string-face This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-voice-lock-type-personality* [Variable]  
Personality used for font-lock-type-face This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-voice-lock-variable-name-personality* [Variable]  
Personality used for font-lock-variable-name-face This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-voice-lock-warning-personality* [Variable]  
Personality used for font-lock-warning-face This personality uses voice-bolden-and-animate whose effect can be changed globally by customizing voice-bolden-and-animate-settings.
- User Option** *emacspeak-w3m-anchor-voice* [Variable]  
Personality used for w3m-anchor This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.
- User Option** *emacspeak-w3m-arrived-anchor-voice* [Variable]  
Personality used for w3m-arrived-anchor This personality uses voice-lighten-extra whose effect can be changed globally by customizing voice-lighten-extra-settings.
- User Option** *emacspeak-w3m-bold-voice* [Variable]  
Personality used for w3m-bold This personality uses voice-bolden-medium whose effect can be changed globally by customizing voice-bolden-medium-settings.
- User Option** *emacspeak-w3m-current-anchor-voice* [Variable]  
Personality used for w3m-current-anchor This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-w3m-form-button-pressed-voice* [Variable]  
Personality used for w3m-form-button-pressed This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-w3m-form-button-voice* [Variable]  
Personality used for w3m-form-button This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.
- User Option** *emacspeak-w3m-form-voice* [Variable]  
Personality used for w3m-form This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-w3m-header-line-location-content-voice* [Variable]  
Personality used for w3m-header-line-location-content This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.

- User Option** *emacspeak-w3m-header-line-location-title-voice* [Variable]  
Personality used for w3m-header-line-location-title This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-w3m-history-current-url-voice* [Variable]  
Personality used for w3m-history-current-url This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-w3m-image-voice* [Variable]  
Personality used for w3m-image This personality uses voice-brighten whose effect can be changed globally by customizing voice-brighten-settings.
- User Option** *emacspeak-w3m-insert-voice* [Variable]  
Personality used for w3m-insert This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-w3m-italic-voice* [Variable]  
Personality used for w3m-italic This personality uses voice-animate whose effect can be changed globally by customizing voice-animate-settings.
- User Option** *emacspeak-w3m-strike-through-voice* [Variable]  
Personality used for w3m-strike-through This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-w3m-tab-selected-voice* [Variable]  
Personality used for w3m-tab-selected This personality uses voice-animate-extra whose effect can be changed globally by customizing voice-animate-extra-settings.
- User Option** *emacspeak-w3m-tab-unselected-voice* [Variable]  
Personality used for w3m-tab-unselected This personality uses voice-monotone whose effect can be changed globally by customizing voice-monotone-settings.
- User Option** *emacspeak-w3m-underline-voice* [Variable]  
Personality used for w3m-underline This personality uses voice-brighten-extra whose effect can be changed globally by customizing voice-brighten-extra-settings.
- User Option** *emacspeak-widget-button-pressed-voice* [Variable]  
Personality used for widget-button-pressed This personality uses voice-bolden-extra whose effect can be changed globally by customizing voice-bolden-extra-settings.
- User Option** *emacspeak-widget-button-voice* [Variable]  
Personality used for widget-button This personality uses voice-bolden whose effect can be changed globally by customizing voice-bolden-settings.
- User Option** *emacspeak-widget-documentation-voice* [Variable]  
Personality used for widget-documentation This personality uses voice-smoothen-extra whose effect can be changed globally by customizing voice-smoothen-extra-settings.
- User Option** *emacspeak-widget-field-voice* [Variable]  
Personality used for widget-field This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.

**User Option** *emacspeak-widget-inactive-voice* [Variable]  
Personality used for widget-inactive This personality uses voice-lighten whose effect can be changed globally by customizing voice-lighten-settings.

**User Option** *emacspeak-widget-single-line-field-voice* [Variable]  
Personality used for widget-single-line-field This personality uses voice-smoothen whose effect can be changed globally by customizing voice-smoothen-settings.

## 13 Emacspeak TTS Servers

Emacspeak produces spoken output by communicating with one of many speech servers. This section documents the communication protocol between the client application i.e. Emacspeak, and the TTS server. This section is primarily intended for developers wishing to:

- Create new speech servers that comply with this communication protocol
- Developers of other client applications who wish to use the various Emacspeak speech servers.

### 13.0.1 High-level Overview

The TTS server reads commands from standard input, and script *speech-server* can be used to cause a TTS server to communicate via a TCP socket. Speech server commands are used by the client application to make specific requests of the server; the server listens for these requests in a non-blocking read loop and executes requests as they become available. Requests can be classified as follows:

- Commands that send text to be spoken.
- Commands that set *state* of the TTS server.

All commands are of the form

```
commandWord {arguments}
```

The braces are optional if the command argument contains no white space. The speech server maintains a *current state* that determines various characteristics of spoken output such as speech rate, punctuations mode etc. (see set of commands that manipulate speech state for complete list). The client application *queues* The text and non-speech audio output to be produced before asking the server to *dispatch* the set of queued requests, i.e. start producing output.

Once the server has been asked to produce output, it removes items from the front of the queue, sends the requisite commands to the underlying TTS engine, and waits for the engine to acknowledge that the request has been completely processed. This is a non-blocking operation, i.e., if the client application generates additional requests, these are processed *immediately*.

The above design allows the Emacspeak TTS server to be *highly* responsive; Client applications can queue large amounts of text (typically queued a clause at a time to achieve the best prosody), ask the TTS server to start speaking, and interrupt the spoken output at any time.

### 13.0.2 Commands That Queue Output.

This section documents commands that either produce spoken output, or queue output to be produced on demand. Commands that place the request on the queue are clearly marked.

```
version
```

Speaks the *version* of the TTS engine. Produces output immediately.

**tts\_say text**

Speaks the specified *text* immediately. The text is not pre-processed in any way, contrast this with the primary way of speaking text which is to queue text before asking the server to process the queue.

**l c**

Speak *c* a single character, as a letter. The character is spoken immediately. This command uses the TTS engine's capability to speak a single character with the ability to flush speech *immediately*. Client applications wishing to produce character-at-a-time output, e.g., when providing character echo during keyboard input should use this command.

**d**

This command is used to *dispatch* all queued requests. It was renamed to a single character command (like many of the commonly used TTS server commands) to work more effectively over slow (9600) dialup lines. The effect of calling this command is for the TTS server to start processing items that have been queued via earlier requests.

**tts\_pause**

This pauses speech *immediately*. It does not affect queued requests; when command *tts\_resume* is called, the output resumes at the point where it was paused. Not all TTS engines provide this capability.

**tts\_resume**

Resume spoken output if it has been paused earlier.

**s**

Stop speech *immediately*. Spoken output is interrupted, and all pending requests are flushed from the queue.

**q text**

Queues *text* to be spoken. No spoken output is produced until a *dispatch* request is received via execution of command *d*.

**a filename**

Cues the audio file identified by *filename* for playing.

**t freq length**

Queues a tone to be played at the specified frequency and having the specified length. Frequency is specified in hertz and length is specified in milliseconds.

**sh duration**

Queues the specified duration of silence. Silence is specified in milliseconds.

### 13.0.3 Commands That Set State

**tts\_reset**

Reset TTS engine to default settings.

**tts\_set\_punctuations mode**

Sets TTS engine to the specified punctuation mode. Typically, TTS servers provide at least three modes:

- None: Do not speak punctuation characters.

- `some`: Speak some punctuation characters. Used for English prose.
- `all`: Speak out *all* punctuation characters; useful in programming modes.

`tts_set_speech_rate rate`

Sets speech rate. The interpretation of this value is typically engine specific.

`tts_set_character_scale factor`

Scale factor applied to speech rate when speaking individual characters. Thus, setting speech rate to 500 and character scale to 1.2 will cause command `l` to use a speech rate of  $500 * 1.2 = 600$ .

`tts_split_caps flag`

Set state of *split caps* processing. Turn this on to speak mixed-case (AKA Camel Case) identifiers.

`tts_capitalize flag`

Indicate capitalization via a beep tone or voice pitch.

`tts_allcaps_beep flag`

Setting this flag produces a high-pitched beep when speaking words that are in all-caps, e.g. abbreviations.

## **14 Acknowledgements.**

Thanks.

## 15 Concept Index

### C

character echo..... 12

### L

line echo ..... 12

### S

speech settings ..... 12

speech system..... 12

### T

tts ..... 12

### W

word echo..... 12

## 16 Key Index

### C

|                                                                                                                                                           |        |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| C-c (.....                                                                                                                                                | 46     |
| C-c ).....                                                                                                                                                | 46     |
| C-c 0.....                                                                                                                                                | 46     |
| C-c 1.....                                                                                                                                                | 46     |
| C-c C-\ .....                                                                                                                                             | 47     |
| C-c C-a .....                                                                                                                                             | 47     |
| C-c C-c.....                                                                                                                                              | 46, 47 |
| C-c C-d.....                                                                                                                                              | 46, 47 |
| C-c C-f .....                                                                                                                                             | 45     |
| C-c C-j .....                                                                                                                                             | 45     |
| C-c C-k .....                                                                                                                                             | 47     |
| C-c C-u .....                                                                                                                                             | 47     |
| C-c C-w .....                                                                                                                                             | 47     |
| C-c C-x C-c.....                                                                                                                                          | 46     |
| C-c C-z .....                                                                                                                                             | 47     |
| C-c e.....                                                                                                                                                | 46     |
| C-c k.....                                                                                                                                                | 46     |
| C-c o.....                                                                                                                                                | 45     |
| control e %.....                                                                                                                                          | 17     |
| control e ' .....                                                                                                                                         | 12     |
| control e / .....                                                                                                                                         | 10     |
| control e =.....                                                                                                                                          | 17     |
| control e [.....                                                                                                                                          | 11     |
| control e {.....                                                                                                                                          | 10     |
| control e 9 control e 8 control e 7 control e 6<br>control e 5 control e 4 control e 3 control<br>e 2 control e 1 control e 0.....                        | 11     |
| control e a.....                                                                                                                                          | 16     |
| control e b.....                                                                                                                                          | 10     |
| control e c.....                                                                                                                                          | 9      |
| control e cap M.....                                                                                                                                      | 16     |
| control e cap R.....                                                                                                                                      | 10     |
| control e cap V.....                                                                                                                                      | 16     |
| control e control @.....                                                                                                                                  | 17     |
| control e control l.....                                                                                                                                  | 17     |
| control e control n.....                                                                                                                                  | 11     |
| control e control o.....                                                                                                                                  | 11     |
| control e control p.....                                                                                                                                  | 11     |
| control e control s.....                                                                                                                                  | 14     |
| control e control w.....                                                                                                                                  | 16     |
| control e d 9 control e d 8 control e d 7<br>control e d 6 control e d 5 control e d 4<br>control e d 3 control e d 2 control e d 1<br>control e d 0..... | 13     |
| control e d a.....                                                                                                                                        | 15     |
| control e d c.....                                                                                                                                        | 13     |
| control e d cap C.....                                                                                                                                    | 13     |
| control e d cap V.....                                                                                                                                    | 15     |
| control e d d.....                                                                                                                                        | 15     |
| control e d f.....                                                                                                                                        | 13     |
| control e d i.....                                                                                                                                        | 14     |
| control e d k.....                                                                                                                                        | 12     |
| control e d l.....                                                                                                                                        | 12     |
| control e d m.....                                                                                                                                        | 13     |
| control e d p.....                                                                                                                                        | 13     |
| control e d q.....                                                                                                                                        | 14     |
| control e d r.....                                                                                                                                        | 13     |
| control e d RETURN.....                                                                                                                                   | 15     |
| control e d s.....                                                                                                                                        | 13     |
| control e d SPACE.....                                                                                                                                    | 15     |
| control e d t.....                                                                                                                                        | 15     |
| control e d w.....                                                                                                                                        | 12     |
| control e d z.....                                                                                                                                        | 15     |
| control e down.....                                                                                                                                       | 10     |
| control e f.....                                                                                                                                          | 16     |
| control e h.....                                                                                                                                          | 17     |
| control e k.....                                                                                                                                          | 17     |
| control e l.....                                                                                                                                          | 10     |
| control e left.....                                                                                                                                       | 11     |
| control e m.....                                                                                                                                          | 16     |
| control e meta control @.....                                                                                                                             | 12     |
| control e n.....                                                                                                                                          | 10     |
| control e p.....                                                                                                                                          | 14     |
| control e r.....                                                                                                                                          | 10     |
| control e right.....                                                                                                                                      | 11     |
| control e SPACE.....                                                                                                                                      | 14     |
| control e t.....                                                                                                                                          | 16     |
| control e up.....                                                                                                                                         | 10     |
| control e v.....                                                                                                                                          | 17     |
| control e w.....                                                                                                                                          | 9      |
| <b>E</b>                                                                                                                                                  |        |
| ESCAPE down.....                                                                                                                                          | 11     |
| ESCAPE next.....                                                                                                                                          | 11     |
| ESCAPE prior.....                                                                                                                                         | 12     |
| ESCAPE up.....                                                                                                                                            | 11     |
| <b>P</b>                                                                                                                                                  |        |
| pause control e s.....                                                                                                                                    | 14     |