Quick Reference for EASE 0.6.0

EASE: The EXAFS Analysis System for Emacs

These pages give a brief description of the commands and user variables available in EASE. EASE is organized into a major mode, INPUT, and a minor mode for FEFF or one of the other programs. The descriptions here are organized similarly. Most of these commands are bound to key sequences as shown here and are also bound to pull-down menu entries. They are presented here in nearly the same order that they are found in the pull down menus. The sub-categories in the table below are mostly the same as the sub-menus in the INPUT and program minor mode menus. Several of the most commonly used commands are also bound to the XEmacs toolbar. These commands are indicated by ⋆.

Symbol guide to key sequences table

`C-`	Hold ⟨ctrl⟩ while hitting the next key
`S-`	Hold ⟨shift⟩ while hitting the next key
`M-`	Hit ⟨esc⟩ then hit the next key
`Mouse-`*n*	Hit the *n*-th mouse button
⋆	Also bound to the XEmacs toolbar

Several commands make use of a *region* of text. A region is defined as the text in between the *point* and the *mark*. The point is the location of the screen cursor. The mark can be set using `M-x set-mark-command` which is bound to `C-`⟨space⟩ and `C-@`. A region may also be painted using `mouse-1`.

EASE comes with support for the programs NORMAL, FLUO, DIFFKK, and PHIT, although this support is not documented in these pages. See the manual for adding support to EASE for other programs and input files.

The latest version of EASE can be found at
`http://feff.phys.washington.edu/~ravel/ease/`

`<ravel@phys.washington.edu>`
Revised 20 May, 2000, printed June 29, 2000

EASE key sequences

INPUT major mode

	key	description
		Editing shortcuts
	`C-c C-s a`	swap true and false at point
	`C-c C-s c`	comment/uncomment current line
	`C-c C-s s`	swap set and guess on current line
	`C-c C-d i`	insert a filename at point
		Clean up
	`C-c C-c l`	tidy up current line
	`C-c C-c r`	tidy up region
	`C-c C-c f`	tidy up entire file
		Keyword functions
	`M-`⟨tab⟩	complete partial keyword
	`M-?`	describe keyword at point
	`M-`⟨ret⟩	verify keyword at point
⋆	`C-c C-b k`	describe all keywords
	`C-c ;`	comment out region
	`C-u C-c ;`	uncomment region
	`M-n`	next hotspot in template
	`M-p`	previous hotspot in template
		Visit files
⋆	`C-c C-f l`	look at log file
	`C-c C-f d`	look at data file at point
	`C-c C-f a`	look at master file
		Set Variables
	`C-c C-d d`	set path to data files
	`C-c C-d f`	set path to FEFF files
	`C-c C-d o`	set path to output files
	`C-c C-d a`	set data, FEFF, and output paths
	`C-c C-d k`	set *k*-weight
	`C-c C-d e`	set E_0 shift
	`C-c C-d m`	set master file
	`C-c C-d v`	set program version
		Run programs
	`C-c C-r a`	run any program on any input file
⋆	`C-c C-r r`	run current program on current file
	`C-c C-r k`	kill running program
		Gnuplot
	`C-c C-p s`	toggle between X11 and PostScript
	`C-c C-f g`	look at GNUPLOT process buffer
	`C-c C-f k`	kill gnuplot process and buffer
		Miscellaneous
	`C-c C-b t`	toggle ease-doc mode
	`C-c C-b b`	submit EASE bug report

continued...

EASE key sequences

	C-c C-b p	submit program bug report
	C-c C-b a	reset variables from Local Variable list
	C-c C-b c	customize EASE
⋆	C-c C-b d	look at program documentation
	C-c C-b e	look at EASE document
	C-c C-b u	look at EASE tutorial
	C-c C-b l	save run-log to a file
	C-c C-b o	look at previous run-log
	C-c C-b s	switch minor modes
	C-c C-b m	display start-up messages
	C-c C-b v	show version EASE version number
	S-Mouse-3	jump to file at point
	M-C-‘	return from jumped-to file
	C-S-l	refresh colorization (hilit19)
	⟨ret⟩	new line and indent

ATOMS minor mode

	key	description
⋆	C-c C-t t	write input file template
⋆	C-c C-r r	run ATOMS
	C-c C-r k	kill ATOMS run
⋆	C-c C-f l	look at ‘feff.inp’
	C-c C-f p	look at ‘p1.inp’
	C-c C-f u	look at ‘unit.dat’
	C-c C-f g	look at ‘geom.dat’
⋆	C-c C-b k	describe ATOMS keywords
⋆	C-c C-e b	evaluate all math expressions in buffer
⋆	C-c C-e l	evaluate math expression on current line

FEFF minor mode

	key	description
⋆	C-c C-t t	write input file template
⋆	C-c C-b k	describe FEFF keywords
		Running
⋆	C-c C-r r	run FEFF
	C-c C-r k	kill FEFF run
	C-c C-s d	toggle value of CONTROL flag
	S-Mouse-3	toggle value of CONTROL flag
		Plotting
⋆	C-c C-p c	plot $\chi(k)$
⋆	C-c C-p x	plot $\mu(E)$ and $\mu_0(E)$
	C-c C-p s	toggle between X11 and PostScript
	C-c C-d k	set k-weight
		Output files
⋆	C-c C-f l	look at output from INTRP
	C-c C-f f	look at ‘files.dat’

continued...

EASE key sequences

	C-c C-f m	look at ‘misc.dat’
	C-c C-f p	look at ‘paths.dat’
	C-c C-f s	look at ‘list.dat’
		Functions for FEFF8
	C-c 8	enable FEFF8 features
⋆	C-c C-e d	plot $\rho(E)$
⋆	C-c C-e x	plot $\mu(E)$
	C-c C-d e	set E_0 shift

AUTOBK minor mode

	key	description
⋆	C-c C-t t	write input file template
⋆	C-c C-f l	look at log file
	C-c C-o n	move to next stanza
	C-c C-o p	move to previous stanza
	C-c C-o m	mark stanza
	C-c C-o k	kill stanza
	C-c C-s p	snag similar from previous stanza
	C-c C-s n	snag similar from next stanza
	C-c C-s e	insert E_0 value from log file
⋆	C-c C-b k	describe AUTOBK keywords
		Running
⋆	C-c C-r r	run AUTOBK
	C-c C-r s	run AUTOBK on current stanza
	C-c C-r k	kill AUTOBK run
		Plotting
⋆	C-c C-p b	plot μ and μ_0, this stanza
⋆	C-c C-p k	plot $\chi(k)$, this stanza
	C-c C-p t	plot $\chi(k)$, data and FEFF
⋆	C-c C-p a	plot all $\chi(k)$ in file
	C-c C-p s	toggle between X11 and PostScript
	C-c C-d k	set k-weight

FEFFIT minor mode

	key	description
		Templates
	C-c C-t f	make ‘feffit.inp’ from ‘files.dat’
	C-c C-t g	make global header template
	C-c C-t l	make local header template
	C-c C-t t	make path paragraph template
	C-c C-t z	make zeroth path template
	C-c C-t b	make background function template
	C-c C-t s	toggle values in background template
⋆	C-c C-b k	describe FEFFIT keywords
		Input and output files
⋆	C-c C-f l	look at log file

continued...

EASE key sequences

⋆	`C-c C-f r`	look at prm file
	`C-c C-f a`	look at master file
	`C-c C-f i`	display output from INTRP
	`C-c C-f f`	look at '`files.dat`'
	`C-c C-f m`	look at '`misc.dat`'
	`C-c C-f p`	look at '`paths.dat`'
	`C-c C-f s`	look at '`list.dat`'
	`C-c C-f t`	make a '`TAGS`' file
		Motion
	`C-c C-o b`	move backward by a paragraph
	`C-c C-o f`	move forward by a paragraph
	`C-c C-o k`	kill the current paragraph
	`C-c C-o m`	mark the current paragraph
	`M-.`	find tag
	`C-x 4 .`	find tag other window
		Editing Shortcuts
	`C-c C-s b`	insert best fits for all guesses
	`C-c C-s g`	insert best fit for guess under point
	`C-c C-s m`	insert McMaster σ^2 from '`feff.inp`'
	`C-c C-s n`	snag similar from next paragraph
	`C-c C-s p`	snag similar from previous paragraph
	`C-c C-c p`	clean up a path paragraph
	`C-c C-c s`	clean up a data set
		Paragraph manipulations
	`C-c C-v i`	reset path index
	`C-c C-v r`	renumber current path paragraph
	`C-c C-v s`	renumber all paragraphs in data set
	`C-u C-c C-v s`	renumber all paragraphs from point
	`C-c C-v a`	add parameters to all paragraphs
	`C-u C-c C-v a`	add parameters from point
	`C-c C-v d`	delete parameters from all paragraphs
	`C-u C-c C-v d`	delete parameters from point
	`C-c C-v c`	comment parameters in all paragraphs
	`C-u C-c C-v c`	comment parameters from point
	`C-c C-v u`	uncomment params in all paragraphs
	`C-u C-c C-v u`	uncomment params from point
		Running
⋆	`C-c C-r r`	run FEFFIT
	`C-c C-r k`	kill FEFFIT run
		Plotting
⋆	`C-c C-p k`	plot data, fit, and marked paths in k
⋆	`C-c C-p r`	plot data, fit, and marked paths in R
⋆	`C-c C-p q`	plot data, fit, and marked paths in q
	`C-c C-p m`	mark or unmark paragraph under point
	`S-mouse-2`	mark or unmark paragraph under mouse
	`C-c C-p a`	mark all paragraphs
	`C-c C-p c`	unmark all paragraph
	`C-c C-p s`	set plot column for R or q

continued...

EASE key sequences

	`C-c C-p s`	toggle between X11 and PostScript
	`C-c C-d k`	set k-weight

GNUPLOT major mode

	key	description
⋆	`C-c C-l`	send line to gnuplot
	`C-c C-v`	send line and move forward 1 line
⋆	`C-c C-r`	send region to gnuplot
⋆	`C-c C-b`	send buffer to gnuplot
	`C-c C-f`	send file to gnuplot
	`C-c C-j`	move to next non-comment line
	`C-c C-i`	insert filename at point
	`C-c C-n`	negate argument
	`M-`⟨tab⟩	complete keyword at point
	`M-`⟨ret⟩	complete keyword at point
	`C-c C-c`	set command arguments with GUI
⋆	`C-c C-e`	look at gnuplot process buffer
	`C-c C-k`	kill gnuplot process and buffer
⋆	`C-c` ⟨up⟩	next script in history list
⋆	`C-c` ⟨down⟩	previous script in history list
	`C-c C-h`	get help from gnuplot document

Using other packages with EASE

ease-doc mode

This is a minor mode for providing on-the-fly descriptions of keywords. When turned on and the point is over a keyword, the description of that keyword will be displayed in the echo area. `C-c C-b t` toggles ease-doc mode on and off.

Math expressions in atoms mode

EASE uses the CALC package to evaluate math expressions for atom coordinates. Variables can be set in lines beginning with `!-` and coordinates can be given as math expressions in lines beginning with `!+`.

Batch processing using dired

Input files can be marked in a dired buffer. `C-c r` then loops through the marked files and runs the program appropriate to each file.

Using Imenu and Speedbar with input files

EASE supplies regular expressions appropriate for use with these packages.

User configurable variables in EASE

There are many variables that the user can set to customize the appearance and behavior of EASE. These can be set in the '`.emacs`' or '`.ease`' file or by using the customize package in recent versions of Emacs and XEmacs. The default values of the variables are given in brackets in the list below. A ♦ means that the default value is too long or too ungainly to print here. In emacs lisp, `nil` is the boolean false value. Any value other that `nil` is considered non-nil. `t` is the boolean true.

ease-base-directory [♦]
Installation location of the EASE source tree. This is determined at the time of installation.

input-bin-location [♦]
Location of the executable scripts and programs that come with EASE. Typically this is relative to `ease-base-directory`.

input-document-location [♦]
Location of the documentation that comes with EASE. Typically this is relative to `ease-base-directory`.

input-program-document-location [♦]
Location of the program documentation used by EASE. Typically this is relative to `ease-base-directory`.

input-glyph-location [♦]
Location of the pixmaps and bitmaps used in the toolbar in XEmacs. Typically this is relative to `ease-base-directory`.

input-comment-delimiter [♦]
This is a long line of equals and plus signs used as decoration separating portions of a '`feffit.inp`' file.

input-stanza-delimiter [♦]
This is a long line of dashes used to separate stanzas.

input-upcase-keywords-flag [`nil`]
Non-nil means to always write keywords in upper case. In FEFF minor mode, this is automatically set to true.

ease-inhibit-startup-message [`nil`]
Non-nil cause EASE to skip its normal sequence of start-up messages.

input-beep-flag [`t`]
Non-nil causes EASE to make noise when it finishes something time consuming.

Variables controlling the interface between EASE and other packages

ease-use-toolbar [`'left`]

Location of toolbar in XEmacs

input-comment-list [♦]
Description of comment string used by the 'comment-out-region' function. This will print a single % followed by a space.

input-mode-variable-comment [`!!&& `]
Comment string used to denote elements of the Local Variables list.

input-prohibit-autoconfig-flag [`nil`]
Non-nil prohibits EASE from automatically writing a Local Variables list

input-emulation [`nil`]
Non-nil means to have EASE automatically invoke emulation software for another editor. Valid values are `vi`, `crisp`, or `edt`

input-document-type [`info`]
Default form of presentation of documentation. The other options are `html` and `text`. Documentation is displayed in an info, w3, or read-only text buffer, as appropriate.

input-time-stamp-flag [`t`]
Non-nil means to automatically apply a time stamp to every input file.

input-time-stamp-begin [♦]

Character string which begins a time stamp.

input-time-stamp-line-limit [`-8`]
Distance from end of file within which the time stamp must be found.

ease-doc-idle-delay [`0.5`]
Number of seconds ease-doc pauses before displaying keyword descriptions.

ease-doc-identifier-string [`*`]
Character appended to `Input` in the modeline to indicate that ease-doc is enabled.

Variables controlling input and output files

input-init-file [`~/.ease`]
Name of the initialization file read when EASE first starts.

input-run-log-interactive [`ease-run.log`]
Default name of file when run-log is saved interactively.

input-run-log [`~/.ease-run.log`]

Name of automatic run-log file

input-run-log-max-lines [`1000`]

Maximum length of run-log file.

input-stanza-name [`ease-stanza.inp`]

Name of input file used for single stanza run.

Variables controlling the appearance of frames

input-use-frames ['own]

Values of 'own or 'share cause EASE to open new windows to display documentation, gnuplot scripts, or the run-log. If this is 'share then the run-log and gnuplot scripts share a frame, otherwise each gets its own. This is set to nil if a non-windowing environment is used.

input-always-raise-flag [t]

Non-nil means to always deiconify and raise a frame when a gnuplot script is written or when a program is executed.

input-doc-frame-plist [♦]

Description of the documentation frame in XEmacs

input-doc-frame-parameters [♦]

Description of the documentation frame in Emacs

input-run-frame-plist [♦]

Description of the run-log frame in XEmacs

input-run-frame-parameters [♦]

Description of the run-log frame in Emacs

input-gnuplot-frame-plist [♦]

Description of the gnuplot frame in XEmacs

input-gnuplot-frame-parameters [♦]

Description of the gnuplot frame in Emacs

Variables controlling indentation and separation

input-stanza-indent [0]

Amount of indentation for lines in a stanza.

input-path-paragraph-indent [0]

Amount of indentation for lines in a path paragraph.

input-path-paragraph-separate [-1]

Amount of separation between columns in a path paragraph.

input-set-guess-indent [0]

Amount of indentation for lines in a set or guess line.

input-set-guess-separate [-1]

Amount of separation between columns in a set or guess line.

input-list-indent [2]

Amount of indentation for lines in a list entry.

input-list-separate [-1]

Amount of separation between columns in a list.

input-feff-indent [1]

Amount of additional indentation for lines in a 'feff.inp' file.

input-potentials-indent [7]

Amount of indentation for lines in a POTENTIALS list in a 'feff.inp' file.

input-potentials-separate [3]

Amount of separation between columns in a POTENTIALS list in a 'feff.inp' file.

input-atoms-separate [3]

Amount of separation between columns in an ATOMS list in a 'feff.inp' file.

input-comment-indent [0]

Amount of indentation for comment lines.

Variables used with gnuplot

gnuplot-script-buffer-name [ease.gp]

Name of buffer containing GNUPLOT script.

input-gnuplot-r-column [4]

Default plot column for plotting $\tilde{\chi}(R)$.

input-gnuplot-q-column [2]

Default plot column for plotting $\tilde{\tilde{\chi}}(k)$.

input-gnuplot-data-style [lines]

Default line type for plots.

input-gnuplot-default-terminal [x11]

Default terminal type for plots. The other option is postscript.

input-gnuplot-default-ps-file [ease.ps]

Default file name for PostScript output.

input-gnuplot-ezero-flag [nil]

Non-nil means to draw a vertical line at E_0 in AUTOBK $\mu(E)$ plots.

ease-gnuplot-history-length [10]

Length of script history list in the GNUPLOT buffer.

input-plot-flag [t]

Non-nil means to always send a newly written script to GNUPLOT. a value of nil may be useful on a graphics incapable terminal.

gnuplot-program [gnuplot]

Name of the GNUPLOT program.

gnuplot-echo-program [t]

Non-nil means to echo every command from a script into a buffer displaying the output of the running GNUPLOT process. This is useful for trouble shooting.

Variables used by particular minor modes

input-best-fit-set-flag [nil]
Non-nil says to swap `guess` for `set` when using `feffit-insert-best-fit` (`C-c C-s b`)

input-intrp-buffer-name [intrp.dat]
Name of buffer to write results from a run of INTRP. Used by FEFF and FEFFIT modes.

input-intrp-args [ ]
Command line arguments passed to INTRP.

input-mcmaster-sigma [sigmm]
Name for the McMaster σ^2 variable used by `Feffit-insert-mcmaster` (`C-c C-s m`)

Feff-8-convergence-filename [convergence.dat]
Default name of file containing convergence data from a FEFF8 run.

Atoms-evaluation-comment-string [!+]
String denoting a line with math expressions describing atom coordinates. This is used in ATOMS mode.

Atoms-definition-comment-string [!-]
String denoting a line variable definitions for math expressions. This is used in ATOMS mode.

Hook Variables in EASE

A *hook* is a variable where you can store a function or functions to be called on a particular occasion.* EASE provides several such variables. See the file '`HOOKS`' which comes with the EASE distribution for example of their use. Always use `(add-hook )` to set a hook variable. Using `(setq )` to do so can have unexpected and undesirable consequences.

* As defined in the emacs lisp reference manual.

input-load-hook
is run when '`input.el`' is loaded.

input-mode-hook
is run when INPUT mode starts in a buffer.

input-before-run-hook
is run when a program in invoked.

input-after-run-hook
is run when a program execution completes.

***program*-load-hook**
is run when '`ease-`*`program`*`.el`' is loaded. There is one of these hooks for each program minor mode.

***program*-mode-hook**
is run when *program* minor mode starts in a buffer. There is one of these hooks for each program minor mode.

ease-doc-load-hook
is run when '`ease-doc.el`' is loaded.

ease-doc-mode-hook
is run when ease-doc minor mode is begun in a buffer.

gnuplot-load-hook
is run when '`gnuplot.el`' is loaded.

gnuplot-mode-hook
is run when GNUPLOT major mode is begun in a buffer.

gnuplot-after-plot-buffer-hook
is run when a full script is sent to GNUPLOT.