

GNU Emacs

Lingfei Kong

lkong@redhat.com

November 13, 2014



Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
 - Key introduce
 - As a editor
 - Programming
- 4 Emacs Other Features
 - GTD, Calendar
 - Browser, picture reader
 - IRC
 - Game
 - info and man document
 - Org-mode
- 5 Reference
- 6 Q & A

Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
- 4 Emacs Other Features
- 5 Reference
- 6 Q & A



redhat

Richard Stallman



Figure 1: Richard Stallman



Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
- 4 Emacs Other Features
- 5 Reference
- 6 Q & A



- Org-mode
- Buffer management, very fast when switch buffers or files
- Have very powerful features
- Can do many things in one emacs session
- Directory Editor
- Bookmark management
- Can configure as a IDE (Python & C & Other language)



- Org-mode
- Buffer management, very fast when switch buffers or files
- Have very powerful features
- Can do many things in one emacs session
- Directory Editor
- Bookmark management
- Can configure as a IDE (Python, C, C++, Other languages)



- Org-mode
- Buffer management, very fast when switch buffers or files
- Have very powerful features
- Can do many things in one emacs session
- Directory Editor
- Bookmark management
- Can configure as a IDE (Python, C, C++, Other languages)



- Org-mode
- Buffer management, very fast when switch buffers or files
- Have very powerful features
- Can do many things in one emacs session
- Directory Editor
- Bookmark management
- Can configure as a IDE (Python & C & Other language)



- Org-mode
- Buffer management, very fast when switch buffers or files
- Have very powerful features
- Can do many things in one emacs session
- **Directory Editor**
- Bookmark management
- Can configure as a IDE (Python & C & Other language)



- Org-mode
- Buffer management, very fast when switch buffers or files
- Have very powerful features
- Can do many things in one emacs session
- Directory Editor
- **Bookmark management**
- Can configure as a IDE (Python & C & Other language)



- Org-mode
- Buffer management, very fast when switch buffers or files
- Have very powerful features
- Can do many things in one emacs session
- Directory Editor
- Bookmark management
- Can configure as a IDE (Python & C & Other language)



Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
- 4 Emacs Other Features
- 5 Reference
- 6 Q & A



redhat

Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
 - Key introduce
 - As a editor
 - Programming
- 4 Emacs Other Features
 - GTD, Calendar
 - Browser, picture reader
 - IRC
 - Game
 - info and man document
 - Org-mode
- 5 Reference
- 6 Q & A



Key introduce

Emacs Key

C = Control

M = Alt = Esc

Del = Backspace

Emacs Configure

`~/.emacs`

`~/.emacs.d`

Install Packages

`M-x package-list-packages`



Key introduce

Emacs Key

C = Control

M = Alt = Esc

Del = Backspace

Emacs Configure

~/.emacs

~/.emacs.d

Install Packages

M-x package-list-packages



Key introduce

Emacs Key

C = Control

M = Alt = Esc

Del = Backspace

Emacs Configure

~/.emacs

~/.emacs.d

Install Packages

M-x package-list-packages



Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
 - Key introduce
 - As a editor
 - Programming
- 4 Emacs Other Features
 - GTD, Calendar
 - Browser, picture reader
 - IRC
 - Game
 - info and man document
 - Org-mode
- 5 Reference
- 6 Q & A



Open, Save, Save as, Close file, Exit emacs I

- C-x C-f: Visit a file ('find-file').
- C-x C-r: Visit a file for viewing, without allowing changes to it ('find-file-read-only').
- C-x C-v: Visit a different file instead of the one visited last
- C-x C-s: Save the current buffer to its file ('save-buffer').
- C-x s: Save any or all buffers to their files ('save-some-buffers').
- C-x C-w: Save the current buffer with a specified file name ('write-file').
- C-x C-c: Offer to save each buffer, then kill the current connection. If the current frame has no client, kill Emacs itself.
- C-x i: Insert contents of file FILENAME into buffer after point. Set mark after the inserted text.
- C-x b: Display buffer BUFFER-OR-NAME in the selected window.



Open, Save, Save as, Close file, Exit emacs II

- C-x C-b: Display a list of existing buffers.
- C-x k: Kill the current buffer.



Buffer

- C-x <LEFT> , C-x <RIGHT>
- M-x rename-buffer
- C-x C-b:
 - . in the first field of a line indicates that the buffer is current. % indicates a read-only buffer. indicates that the buffer is “modified” .
 - d: Flag the buffer for deletion (killing)
 - s: Flag the buffer for saving (Buffer-menu-save)
 - x: Perform all flagged deletions and saves
 - u: Remove all flags from the current line, and move down
 - f/ENTER: Select this line’s buffer in this window.
 - q: Quit buffer list
 - T: Delete, or reinsert, lines for non-file buffers (Buffer-menu-toggle-files-only)



Move course

- C-f, C-b, C-p, C-n: forward; backward; previous line; next line
- M-f, M-b: forward word; backward word
- C-a, C-e: Go to the beginning of the line; Go to the end of the line.
- C-v, M-v: Scroll text of selected window upward ARG lines; Scroll text of selected window down ARG line
- M-<, M->: Move point to the beginning of the buffer; Move point to the end of the buffer



Editing I

- M-n: Repeat n times for the next command
- M-d: Kill characters forward until encountering the end of a word
- C-d: Delete the next character
- C-k: Kill line, to tags or end of line.
- C-Space/C-@: Set the mark at point, and activate it.
- C-w: Kill ("cut") text between point and mark.
- M-w: Save the region as if killed, but don't kill it.
- C-j: Goto next table row or insert a newline and indent.
- C-y: Yank. If the kill is a subtree, treat it specially.
- M-y: Replace just-yanked stretch of killed text with a different stretch.
- C-x C-x: Put the mark where point is now, and point where the mark is now.



Editing II

- C-t, M-t: Interchange characters around point, moving forward one character; Interchange words around point, moving forward one word.
- M-u, M-l, M-c: Convert word to upper case; Convert word to upper case; Convert word to lower case; Capitalize word



Search and Replace

- C-s, C-r: Search forward; Search backward
- M-%: Query and replace
 - .: only replace the current place and exit
 - !: replace all place
 - q: exit
- Find more at [Emacs Keys](#)



Windows

- C-x 2: Split the selected window into two windows, one above the other
- C-x 3: Split the selected window into two side-by-side windows
- C-x o: Select another window in cyclic ordering of windows.
- C-x 0: Delete WINDOW.
- C-x 1: Make WINDOW fill its frame.
- C-x ^: Make the selected window DELTA lines taller.
- M-x shrink-window: Make the selected window DELTA lines smaller.
- M C-v: Scroll the other window
- C-x 4 f: Edit file FILENAME, in another window.



Bookmark I

- C-x r m: Set the bookmark for the visited file, at point.
- C-x r b: Jump to the bookmark named BOOKMARK ('bookmark-jump').
- C-x r l: List all bookmarks ('list-bookmarks').
 - d: delete
 - x: run
 - r: rename
 - s: save
 - f: switch
 - q: quit
 - w: show the current path
 - t: switch to show path
- M-x bookmark-rename: Rename a bookmark.
- M-x bookmark-delete: Delete the bookmark named BOOKMARK.



Bookmark II

- M-x bookmark-save: Save all the current bookmark values in the default bookmark file.
- M-x bookmark-write: Save all the current bookmark values in the file FILENAME.
- M-x bookmark-load: Load a file named FILENAME that contains a list of bookmark values.



Shell

- C-c C-c: Stop to run in shell mode
- M-p, M-n: Show previous command; Show next command
- C-c C-d: Send EOF
- C-c C-z: Like C-z in bash shell



Directory Editor I

- C-x d:
 - C: copy
 - d: ready to delete
 - D: delete immediately
 - f: open file or directory
 - g: refresh
 - n, p: move next line; move previous line
 - k: kill from the screen
 - o: open it in the next window and focus cursor in it
 - C-o: open it in the next window but not focus cursor in it
 - q: quit dired
 - R: rename file name
 - u: remove flag
 - v: show file content in a read-only mode



Directory Editor II

>: move to next directory
<: move to previous directory
s: switch sort mode



Help

- C-h t: Select the Emacs learn-by-doing tutorial.
- C-h i: Enter Info, the documentation browser.
- C-h C-f: Display the Emacs Frequently Asked Questions (FAQ) file.
- C-h p: Find packages matching a given keyword.
- C-h k: Display documentation of the function invoked by KEY.
- C-h f: Display the full documentation of FUNCTION (a symbol).
- C-h v: Display the full documentation of VARIABLE (a symbol).
- C-h b: Show a list of all defined keys, and their definitions.
- C-h m: Check the enabled mode for current buffer
- C-h l: Display last 300 input keystrokes.
- C-h c: Print the name of the function KEY invokes.
- C-h i d m ecb RET i topic RET



Macro

- C-x (: Record subsequent keyboard input, defining a keyboard macro.
- C-x): Finish defining a keyboard macro.
- C-x e: Call last keyboard macro, ending it first if currently being defined.
- M-x name-last-kbd-macro: Assign a name to the last keyboard macro defined.
- M-x insert-kbd-macro: Insert in buffer the definition of kbd macro NAME, as Lisp code.
- M-x load-file: Load the Lisp file named FILE.



Others

- C-g: Signal a 'quit' condition.
- M-x revert-buffer: Replace current buffer text with the text of the visited file on disk.
- M-x: recover-file: Visit file FILE, but get contents from its last auto-save file.
- M-x: recover-session: Recover auto save files from a previous Emacs session.



Agenda

- 1 Author
- 2 Why Emacs
- 3 **Emacs Basic Features**
 - Key introduce
 - As a editor
 - **Programming**
- 4 Emacs Other Features

- GTD, Calendar
- Browser, picture reader
- IRC
- Game
- info and man document
- Org-mode

5 Reference

6 Q & A



Python

- C-M-a: py-beginning-of-def-or-class
- C-M-e: py-end-of-def-or-class
- C-M-h: py-mark-def-or-class
- C-c #: py-comment-region
- C-c ?: py-describe-mode
- C-c <tab>: Reindent a region of Python code.



Shell

- C-c C-c: case statement
- C-c C-f: for loop
- C-c (: function definition
- C-c TAB: if statement
- C-c C-l: indexed loop from 1 to n
- C-c C-o: while getopts loop
- C-c C-r: repeat loop
- C-c C-s: select loop
- C-c C-u: until loop
- C-c C-w: while loop



Configure as a python IDE

- See [Configure emacs as a Python IDE](#)
- Configure ECB
- Create etags files

```
find . -name "*.py" -print | etags -
```

- Selete a Tags table in emacs
- M-x visit-tags-table



Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
- 4 Emacs Other Features
- 5 Reference
- 6 Q & A



redhat

Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
 - Key introduce
 - As a editor
 - Programming
- 4 Emacs Other Features

- GTD, Calendar
- Browser, picture reader
- IRC
- Game
- info and man document
- Org-mode

5 Reference

6 Q & A



GTD, Calendar I

- Shift + <-/-> to change status of a Item, Shift + Up/Down to change priority
- C-c C-t: Switch event
- C-c c: Capture something (Use C-c C-c to quit)
- C-c a: Dispatch agenda commands to collect entries to the agenda buffer.
- C-c Search tags
- C-c C-c: Add tags
- C-c / t: Search todo list
- C-c a t: global TODO list
- C-c ,: set priority
- C-c C-d: set deadline



GTD, Calendar II

- C-c [: add current file to agend
- C-c]: remove current file to agend
- Have a look a calendar



redhat

Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
 - Key introduce
 - As a editor
 - Programming
- 4 Emacs Other Features
 - GTD, Calendar
 - **Browser, picture reader**
 - IRC
 - Game
 - info and man document
 - Org-mode
- 5 Reference
- 6 Q & A



Browser, picture reader

- Try to open picture in emacs

- Show w3m in emacs

g: w3m-goto-url

B: w3m-view-previous-page

N: w3m-view-next-page

<: w3m-scroll-right

>: w3m-scroll-left

H: w3m-gohome

I: w3m-view-image

More about emacs-w3m, see: [Emacs w3m 操作快捷键](#)



Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
 - Key introduce
 - As a editor
 - Programming
- 4 Emacs Other Features
 - GTD, Calendar
 - Browser, picture reader
 - **IRC**
 - Game
 - info and man document
 - Org-mode
- 5 Reference
- 6 Q & A



IRC

- Show erc: `irc.devel.redhat.com`

`/list`

`/join`

`/names`

`/quit reason`

`/away reason`

`/whois nickname`

`/whoami`

`/nick newname`

`/msg nickname`

`/query nickname`

`/whois`

For more keys see: [IRC/ERC commands](#)



Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
 - Key introduce
 - As a editor
 - Programming
- 4 Emacs Other Features
 - GTD, Calendar
 - Browser, picture reader
 - IRC
 - **Game**
 - info and man document
 - Org-mode
- 5 Reference
- 6 Q & A



Game

- Show game: snake
- More games: gomoku, pong, etc



redhat

Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
 - Key introduce
 - As a editor
 - Programming
- 4 Emacs Other Features
 - GTD, Calendar
 - Browser, picture reader
 - IRC
 - Game
 - info and man document
 - Org-mode
- 5 Reference
- 6 Q & A



info and man document

- M-x man ENTER ls



redhat

Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
 - Key introduce
 - As a editor
 - Programming
- 4 Emacs Other Features
 - GTD, Calendar
 - Browser, picture reader
 - IRC
 - Game
 - info and man document
 - **Org-mode**
- 5 Reference
- 6 Q & A



Org-mode

- Org mode is a variant of Outline mode for using Emacs as an organizer and/or authoring system.
- Export
- Note and Presentation
- GTD
- Blog
- See more at official website for [Org mode](#)



Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
- 4 Emacs Other Features
- 5 **Reference**
- 6 Q & A



redhat

Reference

- [Configuring Emacs as a Python IDE](#)
- [Writing Beamer presentations in org-mode](#)
- [Writing Non-Beamer presentations in org-mode](#)
- [GNU Emacs](#)
- [My Emacs configure](#)
- [Org Mode - Organize Your Life In Plain Text!](#)
- [用 Org-mode 实现 GTD](#)
- [Org for GTD and other Task managment systems](#)



Agenda

- 1 Author
- 2 Why Emacs
- 3 Emacs Basic Features
- 4 Emacs Other Features
- 5 Reference
- 6 Q & A



redhat

Thanks for your attention!

Any questions?