Emacs

Instructor: Yongjie Zheng
March 15, 2016

Emacs

- Fundamentals

- Emacs Command Structure
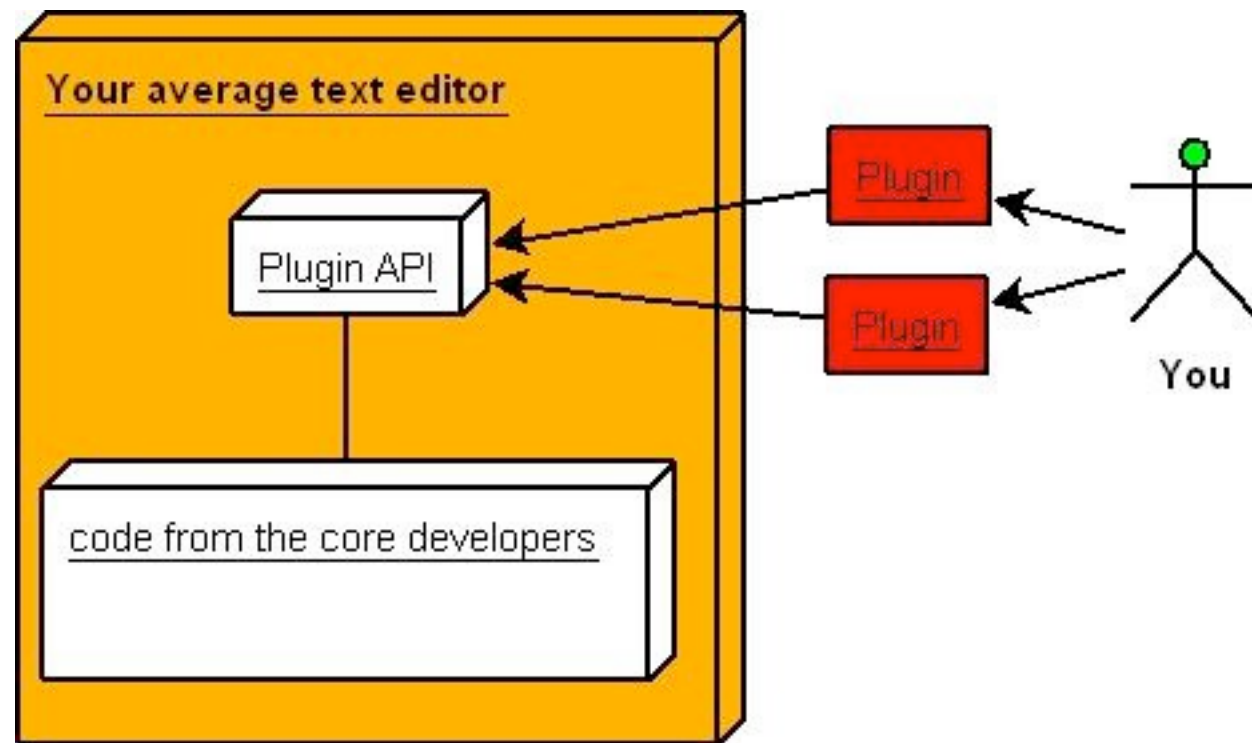
- Basic Commands

Emacs Fundamentals

- Emacs is a programmer's editor

  - Emacs is not WYSIWYG.

- Emacs is free, open-source, and can run on most major operating systems.

- Emacs history, from Wikipedia.

  - First developed by Richard Stallman and Guy L. Steele, Jr. in 1976.

  - The most popular, and most ported version of Emacs is GNU Emacs from the GNU project.

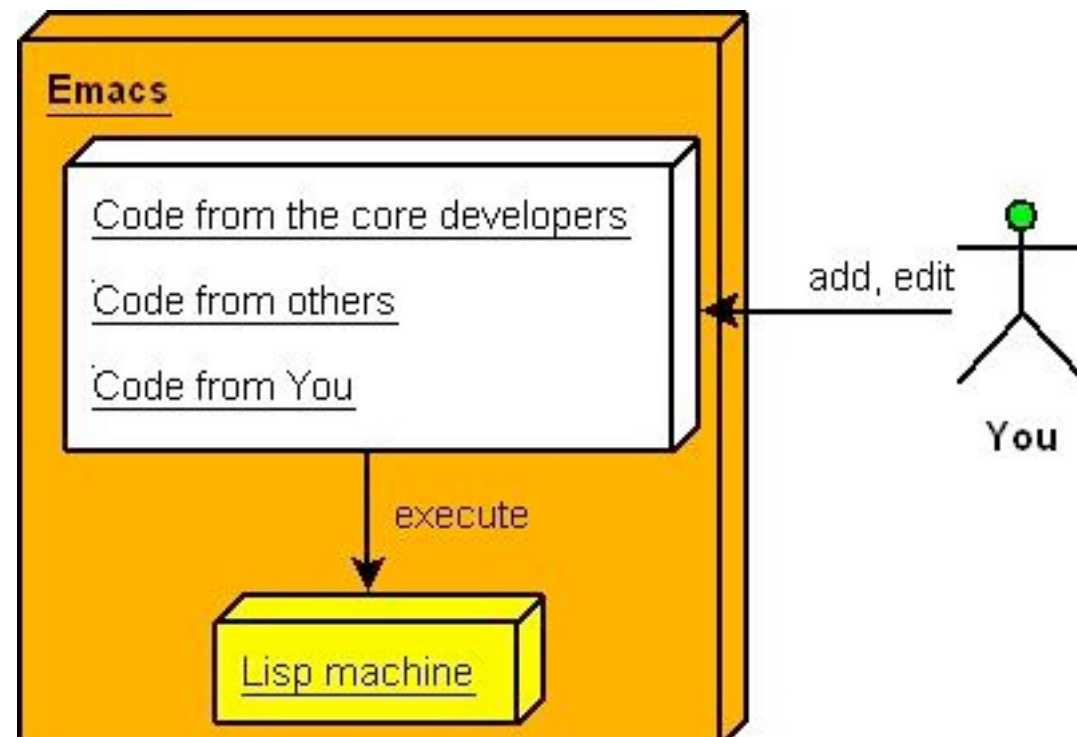 - XEmacs is a common variant that branched from GNU Emacs in 1991.

Emacs Fundamentals, cont.

- In general, Emacs consists of

    1. Emacs lisp machine (written in C) that interprets the language of Emacs Lisp.

    2. Emacs editor: implemented in Emacs Lisp and this is what people usually mean by "Emacs".

    3. The set of packages included in a default Emacs installation.

    4. Your own code that adds additional functionalities to Emacs.

- Emacs is highly customizable and extendable.

    - You can change or redefine 2, 3, and 4 above for your own purpose.

Emacs v.s Other Editors (e.g. Eclipse)

Your average text editor

Plugin

Plugin API

Plugin

You

code from the core developers

Other Editor's Architecture

Emacs

Code from the core developers

Code from others

Code from You

add, edit

You

execute

Lisp machine

Emacs Architecture

Emacs Fundamentals, cont.

- Entering and Exiting Emacs

  - To enter: emacs

  - To open a file (e.g. HW.java) with emacs: emacs HW.java

  - To exit: C-x C-c

- Notations

  - C-x: For any x, the character Control-x.

  - M-x: For any x, the character Meta-x.

    - Meta by default may be bound to different keys (e.g. Alt, Esc, Command).

  - C-M-x: For any x, the character Control-Meta-x.

Emacs Fundamentals, cont.

- File: an actual file on disk. A copy of a file can be loaded into Emacs to initialize a buffer; a copy of a buffer can be written to a file to save it.

- Buffer: the internal data structure that holds the text you actually edit. Emacs can have any number of buffers active at any moment.

- Window: your view of a buffer. Windows can be created and deleted at will.

Emacs Fundamentals, cont.

- Every buffer has a major mode, and may have zero or more minor modes.

- Emacs chooses a major mode for you automatically, typically based on a file extension.

- You can always set the mode explicitly.

  - M-x c mode, M-x fundamental-mode, etc.

- Mode alters behavior, key bindings, and text display.

- Some major modes: Fundamental Mode, Text Mode, C Mode, etc.

Emacs Screen

- The mode line (near the bottom of the screen) displays the information about

  - The state of the buffer, one of modified (indicated by a pair of asterisks), unmodified (hyphens), or read-only (indicated by a pair of % signs).

  - The name of the file you are editing (it will be *scratch* if you are not editing any file).

  - The amount of the file that you can see on the screen:

    - All - You can see all of the file; Top - You can see the top of the file; Bot - You can see the bottom of the file.

  - The major mode (in parenthesis).

- The Minibuffer (echo area): the blank area below the mode line.

Backup and Autosave

- Backup files: Emacs always saves the previous version of your file when you save. If your file is named foo, the backup will be called foo~.

- Auto-save files: Emacs, by default, auto-saves your file while you are editing it. The auto-save file for a file foo is called #foo#. If Emacs (or the system) were to crash before you could save your edits, you can recover almost all of it from this file. Auto-saving happens (by default) every 300 characters, or when a system error is encountered.

Emacs Command Structure

- Emacs command

  - All Emacs commands are entered as 8-bit ASCII characters.

  - Every command has a long name, such as kill-line, delete-backward-char.

  - Key binding: pairing of keystroke and command

  - Emacs command set: the set of all bindings

- About ASCII

  - 7 bit ASCII: 95 printable characters + 32 control characters + DEL (the delete character)

  - 8-bit ASCII: with the high-order bit set

Dec	Hx	Oct	Char			Dec	Hx	Oct	Html	Chr		Dec	Hx	Oct	Html	Chr		Dec	Hx	Oct	Html	Chr
0	0	000	NUL	(null)		32	20	040	 	Space		64	40	100	@	@		96	60	140	`	`
1	1	001	SOH	(start of heading)		33	21	041	!	!		65	41	101	A	A		97	61	141	a	a
2	2	002	STX	(start of text)		34	22	042	"	"		66	42	102	B	B		98	62	142	b	b
3	3	003	ETX	(end of text)		35	23	043	#	#		67	43	103	C	C		99	63	143	c	c
4	4	004	EOT	(end of transmission)		36	24	044	$	$		68	44	104	D	D		100	64	144	d	d
5	5	005	ENQ	(enquiry)		37	25	045	%	%		69	45	105	E	E		101	65	145	e	e
6	6	006	ACK	(acknowledge)		38	26	046	&	&		70	46	106	F	F		102	66	146	f	f
7	7	007	BEL	(bell)		39	27	047	'	'		71	47	107	G	G		103	67	147	g	g
8	8	010	BS	(backspace)		40	28	050	(	(		72	48	110	H	H		104	68	150	h	h
9	9	011	TAB	(horizontal tab)		41	29	051	)	)		73	49	111	I	I		105	69	151	i	i
10	A	012	LF	(NL line feed, new line)		42	2A	052	*	*		74	4A	112	J	J		106	6A	152	j	j
11	B	013	VT	(vertical tab)		43	2B	053	+	+		75	4B	113	K	K		107	6B	153	k	k
12	C	014	FF	(NP form feed, new page)		44	2C	054	,	,		76	4C	114	L	L		108	6C	154	l	l
13	D	015	CR	(carriage return)		45	2D	055	-	-		77	4D	115	M	M		109	6D	155	m	m
14	E	016	SO	(shift out)		46	2E	056	.	.		78	4E	116	N	N		110	6E	156	n	n
15	F	017	SI	(shift in)		47	2F	057	/	/		79	4F	117	O	O		111	6F	157	o	o
16	10	020	DLE	(data link escape)		48	30	060	0	0		80	50	120	P	P		112	70	160	p	p
17	11	021	DC1	(device control 1)		49	31	061	1	1		81	51	121	Q	Q		113	71	161	q	q
18	12	022	DC2	(device control 2)		50	32	062	2	2		82	52	122	R	R		114	72	162	r	r
19	13	023	DC3	(device control 3)		51	33	063	3	3		83	53	123	S	S		115	73	163	s	s
20	14	024	DC4	(device control 4)		52	34	064	4	4		84	54	124	T	T		116	74	164	t	t
21	15	025	NAK	(negative acknowledge)		53	35	065	5	5		85	55	125	U	U		117	75	165	u	u
22	16	026	SYN	(synchronous idle)		54	36	066	6	6		86	56	126	V	V		118	76	166	v	v
23	17	027	ETB	(end of trans. block)		55	37	067	7	7		87	57	127	W	W		119	77	167	w	w
24	18	030	CAN	(cancel)		56	38	070	8	8		88	58	130	X	X		120	78	170	x	x
25	19	031	EM	(end of medium)		57	39	071	9	9		89	59	131	Y	Y		121	79	171	y	y
26	1A	032	SUB	(substitute)		58	3A	072	:	:		90	5A	132	Z	Z		122	7A	172	z	z
27	1B	033	ESC	(escape)		59	3B	073	;	;		91	5B	133	[	[		123	7B	173	{	{
28	1C	034	FS	(file separator)		60	3C	074	<	<		92	5C	134	\	\		124	7C	174	|		
29	1D	035	GS	(group separator)		61	3D	075	=	=		93	5D	135	]	]		125	7D	175	}	}
30	1E	036	RS	(record separator)		62	3E	076	>	>		94	5E	136	^	^		126	7E	176	~	~
31	1F	037	US	(unit separator)		63	3F	077	?	?		95	5F	137	_	_		127	7F	177	&#127;	DEL

12

Simple Commands

- self-insert-command: 95 printable ASCII characters.

- 32 editing commands: 32 control characters (E.g. C-a, C-b).

- Another 128 editing commands: characters in the upper half of ASCII with the high-order bit set.

  - Use the meta key, such as M-a, M-b, for printable characters, and C-M-a, for the control characters.

Prefix (Compound) Commands

- Emacs has more than 160 commands!

- A prefix command makes a pair of keystrokes (typed consecutively, not simultaneously) bound to one command.

- The standard prefix commands are:

  - C-c: used for commands that are specific to particular modes.

  - C-h: used for Help commands.

  - C-x: Character eXtend. Followed by one character.

  - M-x: Named command eXtend. Followed by a long name.

Too Many Commands?

- In practice, you may only need a small potion of these commands.

- You learn new commands all the time.

- Getting help with Emacs

  - C-h a: Prompts for a keyword and then lists all the commands with that keyword in their long name.

  - C-h k: Prompts for a keystroke and describes the command bound to that key, if any.

  - C-h m: Describes the current major mode and its particular key bindings.

  - C-h t: run the Emacs tutorial.

Basic Commands

- Undo and Quit

- Commands to Manipulate Files/Buffers/Windows

- Moving Around in Buffers

- Searching and Replacing

- Deleting, Killing, and Yanking

- Copying and Moving Text

- The Mark and The Region

Undo and Quit

- Undo: C-_ or C-x u

- Cancel: C-g

- M-x shell: run Unix shell

Commands of Files/Buffers/Windows

- Commands to Manipulate Files

  - C-x C-f: read a file into a buffer for editing.

  - C-x C-s: save a file.

  - C-x s: save all the buffers.

- Commands to Manipulate Buffers

  - C-x b: switch to another buffer.

  - C-x C-b: pop up a new window that lists all your buffers.

  - C-x k: kill a buffer.

- Commands to Manipulate Windows

  - C-x 0: close the current window.

  - C-x 1: delete all other windows except the current one.

  - C-x 2, C-x 3: split the current window in two (vertically, horizontally).

  - C-x o: switch to another window.

Moving Around in Buffers

- Move to the beginning, end of the current buffer: C-x [, C-x]

- Move to the beginning, end of the current buffer (with the mark set to where you were): M-<, M->

- Page down and up: C-v, M-v

- Go to specific line of number: M-g g

Searching and Replacing

- Search for a text

  - Search: C-s TEXT

  - Search for next occurrence: C-s

  - Stop at found occurrence: RET (Return Key)

  - C-r to search backwards

- Search and replace, asking for confirmation

  - M-%: prompts you for the text to replace, and the text to replace it with, and then searches and replaces within the current buffer.

  - For each match, you have the following options:

    - SPC (Space Key): perform this replacement.

    - DEL (Delete Key): don't perform this replacement.

Deleting, Killing, and Yanking

- Deleting

- Killing (Cutting)

  - C-k: Kills to the end of the current line, not including the newline.

  - M-d: Kills to the end of the word to the right of the cursor (forward).

  - M-DEL: Kills to the beginning of the word to the left of the cursor (backward).

- Yanking (Pasting)

  - The kill ring: almost all commands which delete text save it for possible later retrieval.

  - C-y: yank earlier killed text.

Copying and Moving Text

- To copy text, kill it, yank it back immediately.

- To move text, just kill it and yank it back elsewhere.

The Mark and the Region

- The region is the text between point and mark.

  - Point is just where the cursor is.

  - The mark, on the other hand, is set with a special command C-SPC (set-mark-command).

- Related commands

  - M-w: copy region without killing.

  - C-w: kill region.

  - C-x h: set the region around the entire buffer.

To summarize: some popular commands

- C-x C-f open a file; C-x C-s: save the file; C-x C-c: exit Emacs; C-x b: new buffer; C-x k: kill buffer.

- C-v, M-v: page down and page up.

- C-s TEXT: search text, C-s (forward) and C-r (backward).

- M-%: find and replace, SPC (replace) and DEL (skip).

- C-k: kill (cut) the line; M-d: kill (cut) the word; C-y: yank (paste) the killed text.

- C-SPC: set the mark; M-w: copy region without killing; C-w: kill region.

- C-x u: undo changes made to the buffer.

- C-g: quit (revoke previous commands).

Reference

- Emacs command structure: A Tutorial Introduction to GNU Emacs http://www2.lib.uchicago.edu/keith/tcl-course/emacs-tutorial.html

- About Emacs architecture: http://blog.bookworm.at/2007/03/introduction-to-all-these-emacs.html

- Some other resources: http://stuff.mit.edu/iap/2009/emacs/