

Introduction to Emacs/Slime/Quicklisp

黄 澗石 (Jianshi Huang)

November 27, 2011

- How to establish a Common Lisp development environment
 - Tutorial
 - SBCL
 - Quicklisp + Slime
 - Emacs is a prerequisite
 - In minimal steps

Step 1

- Install Emacs

- Install SBCL
 - install from pre-compiled binary
<http://www.sbcl.org/platform-table.html>
 - install from git repo (need an installed implementation)
 - `git clone git://sbcl.git.sourceforge.net/gitroot/sbcl/sbcl.git`
 - check NEWS
 - `git checkout <latest stable release>`
 - `./clean`
 - `./make.sh`
 - `cd tests && sh ./run-tests.sh`

- Install Quicklisp
 - go to <http://www.quicklisp.org/> and download quicklisp.lisp
 - in SBCL, load quicklisp.lisp
 - (quicklisp-quickstart:install)
 - (ql:add-to-init-file)
 - check ~/.sbclrc

Step 4

- Install quicklisp-slime-helper
 - (ql:quickload :quicklisp-slime-helper)
 - add configurations to ~/.emacs
 - restart Emacs

- M-x slime

- <http://www.emacswiki.org/>
- disable toolbar and scrollbar

```
1 (tool-bar-mode -1)
2 (scroll-bar-mode -1)
```

- swap () and [] keys

```
1 (define-key key-translation-map [?\()] [?\[])
2 (define-key key-translation-map [?\[] [?\()])
3 (define-key key-translation-map [?\)] [?\]])
4 (define-key key-translation-map [?\]]) [?\)])
```

- paredit for paired paren edit

```
1 sudo aptitude install paredit-el
2
3 (require 'paredit)
4 (add-hook 'emacs-lisp-mode-hook      (lambda () (paredit-mode 1)))
5 (add-hook 'lisp-mode-hook           (lambda () (paredit-mode 1)))
6 (add-hook 'lisp-interaction-mode-hook (lambda () (paredit-mode 1)))
7 (add-hook 'scheme-mode-hook         (lambda () (paredit-mode 1)))
```


- paren-face for reducing color

```
1 cd ~/.emacs.d && wget http://www.davep.org/emacs/parenface.el
2 (require "parenface")
3 (set-face-foreground 'paren-face "gray50")
```

- paren matching

```
1 (show-paren-mode 1)
2 (setq show-paren-delay 0)
```

- mark sexp

```
1 (define-key global-map (kbd "C-M-h") 'mark-sexp)
```

• cursor movement

```
1 (define-key global-map (kbd "C-f") 'forward-sexp)
2 (define-key global-map (kbd "M-f") 'forward-word)
3 (define-key global-map (kbd "C-M-f") 'forward-list)
4 (define-key global-map (kbd "C-b") 'backward-sexp)
5 (define-key global-map (kbd "M-b") 'backward-word)
6 (define-key global-map (kbd "C-M-b") 'backward-list)
```

• kill

```
1 (define-key global-map (kbd "C-k") 'kill-sexp)
2 (define-key global-map (kbd "M-k") 'kill-line)
3 (define-key global-map (kbd "C-M-k") 'kill-sentence)
```

- character encoding

```
1 (set-language-environment "UTF-8")  
2 (setq slime-net-coding-system 'utf-8-unix)
```

- coding style

```
1 (define-common-lisp-style "mine"  
2   (:inherit "modern")  
3   (:variables  
4     (lisp-loop-indent-subclauses t)))  
5  
6 (setq common-lisp-style-default "mine")
```

- slime selector

```
1 (define-key global-map (kbd "C-c s") 'slime-selector)
```

Slime Tips (Cont.)

- cldoc

```
1 (load-library "cldoc")  
2 (setq cldoc-argument-case 'downcase  
3       cldoc-idle-delay 0.2)
```

- multilined autodoc

```
1 (setq slime-autodoc-use-multiline-p t)
```

- temporary fasl mode

```
1 (setq slime-compile-file-options '(:fasl-directory "/tmp/slime-fasls/"))  
2 (make-directory "/tmp/slime-fasls/" t)
```

Slime Tips (Demo)

- buffers
- compilation and notes
- completion
- finding definitions
- cross references
- macro expansion
- debugging
- disassembly
- inspection
- tracing
- profiling

Slime Tips (Remote)

- create server
 - in ~/.sbclrc

```
1 (swank:create-server :port 4005 :dont-close t :coding-system "utf-8-unix")
```

- globally-redirect-io
 - in ~/.sbclrc

```
1 (setf swank:*globally-redirect-io* t)
```

Slime Tips (More commands)

- useful but uncommon keys

```
1 C-c C-y      slime-call-defun
2 C-c C-s      slime-complete-form
3 C-c M-o      slime-repl-clear-buffer
4 C-c C-v M-o  slime-clear-presentations
5 C-c E        slime-edit-value
6 C-c x        slime-export-symbol-at-point
7 C-c ~        slime-sync-package-and-default-directory
8 C-u C-c C-c  compile for debug
9 C-u C-c C-k  compile for debug
10 M-- C-c C-c compile for speed
11 M-- C-c C-k compile for speed
12 C-c C-v d    slime-describe-presentation-at-point
13 C-c C-v i    slime-inspect-presentation-at-point
```

Slime Tips (More commands)

- useful commands (no default keys)

```
1 slime-browse-classes
2 slime-browse-xrefs
3 slime-macroexpand-1-inplace
4 slime-insert-balanced-comments
5 slime-compiler-macroexpand-1
6 slime-compiler-macroexpand
7 slime-inspect-definition
8 slime-disassemble-definition
```

- useful selector commands

```
1 ,l          slime-load-system
2 ,c          compile-system but not load
3 ,o          slime-open-system
4 ,b          slime-browse-system
5 ,f-c        force-compile-system
6 ,f-l        force-load-system
```


- Emacs Wiki
<http://www.emacswiki.org/>
- Slime
<http://common-lisp.net/project/slime/>
- Quicklisp
<http://www.quicklisp.org/>
- Slime Tips
<http://slime-tips.tumblr.com/>
 - or google 'slime tips'
- CLiki
<http://www.cliki.net/SLIME>