Tools Talk

Emacs

A Text Editor and More

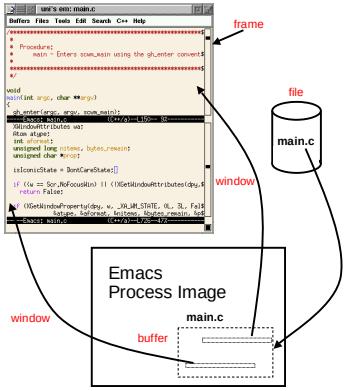
Michael Ernst & Greg J. Badros
University of Washington
Computer Science & Engineering
5 October 1998

http://www.cs.washington.edu/uns/ToolsTalks

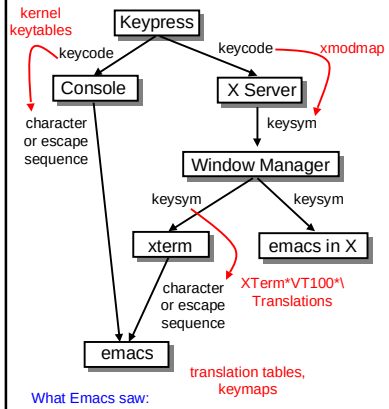
Outline

- Basic concepts
- How to get more help
- Customizing Emacs
- Emacs usage tips
- Packages (extensions)
- Stump the experts

Files, Buffers, Windows, and Frames



Keystroke handling



Getting help

- `M-x help` or `C-h`
- Emacs tutorial: `C-h t`
- Online manuals: `C-h i` or `M-x info`
  - Emacs, extension packages
  - Emacs Lisp, E-Lisp intro
  - Info, much more
- FAQ: /uns/share/doc/emacs-faq
- Package source
  /uns/share/emacs/19.34/lisp/
  /uns/share/emacs/20.3/lisp/
  /uns/share/xemacs/20.4/lisp/
  /uns/share/emacs/site-lisp/

Getting more help

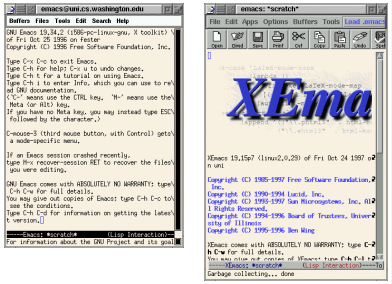
- `M-x describe-mode (C-h m)`
- Commands relating to a task
  - all packages by keyword: `C-h p`
  - `M-x apropos` or `C-h a`
  - `M-x lisp-dir-apropos`
- `C-h`: help
  - **b**indings (for keys)
  - **f**unctions
  - **v**ariables
  - after some other keys

Emacs versions

Emacs 19: `emacs` (default version)
Emacs 20: `emacs-20` (Mike recommends)
XEmacs 20: `xemacs` (Greg recommends)

- For casual users, any /uns version
- Emacs: more stable and consistent
- XEmacs: more features
  - Windows: widgets, fonts, toolbar, graphics, embedding; non-X faces
  - gnuattach: connect to server from a terminal
  - More bundled packages, APIs, ToolTalk (compiler integration, Emacs Lisp, customization)

Emacs and XEmacs



Comon Startup Options

`-q`	Do not load ~/.emacs
`-debug-init`	Enable debugger during initialization (useful to debug ~/.emacs)
`-u` *user*	Load ~*user*/.emacs instead of yours
`--no-site-file`	Do not load /uns/share/emacs/site-start.el
`-nw`	Ignore DISPLAY and use tty
`-display`	Choose X server for display
`-l` *file*	Load *file* after startup
`-f` *func*	Run *func* after startup
`--batch`	Do not run interactively (useful for just byte-compiling)

Customization

- `M-x customize` ([X]Emacs 20)
- Setting variables

```
(setq mail-yank-prefix "> ")
(setq require-final-newline t)
(setq auto-mode-alist
      (cons '("\\.lsp" . common-lisp-mode)
            auto-mode-alist))
(setq load-path
      (cons (expand-file-name "~/emacs")
            load-path))
(setq font-lock-string-face
      'midnightblue)
```

- Using packages

```
(autoload 'mc-install-read-mode
  "mailcrypt" nil t)
(require 'uniquify)
(load "mode-hooks-mde")
```

See the instructions at the top of the source file

More customization

- Binding keys

```
(global-set-key "\C-l" 'smart-recenter)
(define-key lisp-mode-map "\C-cl" 'run-
  lisp)
```

- Mode hooks

```
(add-hook 'text-mode-hook
          'auto-fill-mode)
(add-hook 'LaTeX-mode-hook
          'auto-fill-mode)
(add-hook 'emacs-lisp-mode-hook
       'turn-on-font-lock)
(global-font-lock-mode t)
```

- Emacs Lisp in ~/.emacs:
  powerful, may be complex;
  cut and paste from other users

Emacs techniques

- Undo: `C-/` or `C-_` or `C-x u`
  buffer modification mark
- `ESC` and `M-` prefixes
- X cut and paste
- Buffer list
- Completion
- Minibuffer history: `M-n` and `M-p`
- Syntax-based editing:
  character, word, sentence, paragraph, line, s-expression, function
  c-beginning-of-statement
  transpose-words

More tricks

- Regular expression search, replace metacharacers: `\(`, `\)`, `\|`
- Narrow to region
- Repetition: `C-u` or `C-7`
- Keyboard (editor) macros
  `C-x(` to start, `C-x)` to end
  {call,edit,name}-last-kbd-macro
  `C-u 0` to repeat until error
- abbrev mode; dynamic abbreviations (`M-/`; “dabbrev”)
- Set auto-save-directory to `/tmp` (especially for Linux)

Extension packages

- Text: mail, news, ispell
- Programming: language modes, compilation, tags, debuggers, interaction
- OS: dired, ftp, crypt, shell
- Version control, diff
- Server
- Amusements

Use familiar editing commands, fully customizable

May be slower in in machine time

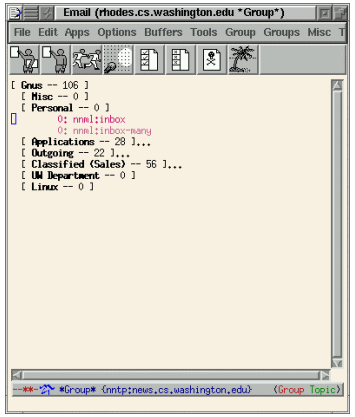
Mail and news

Rmail (Emacs only)
VM (XEmacs only)
MH-E

Gnus

sorting, scoring, folders

Gnus (netnews and mail)



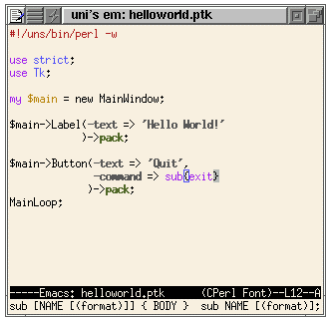
Spellchecking with ispell

- `M-x spell-buffer`
- `(add-hook 'mail-send-hook 'ispell-message)`
- `M-$` or `M-x spell-word`
- Per-file local-words annotation

Language major modes

- Most languages, including
  - cc-mode: C, C++, Objective C, Java
  - cperl-mode: Perl
  - AUC Tex: LaTeX, TeX, BibTeX, ...
- Features
  - indentation
  - syntax-related movement
  - colorization
  - documentation links
  - execution

Cperl mode



Compilation

- `C-c C-c` or `M-x compile`
- `C-x `` or `M-x next-error` to iterate through errors
- Errors in `*compilation*` buffer are clickable
- `M-x grep` is similar

Debugging

- `M-x gdb`
- Usual gdb commands
- In source file, set breakpoints and more
- Supported debuggers:
  gdb, sdb, dbx, xdb, perldb, pdb, jdb

Interpreter interaction:
`M-x run-lisp`, `M-x py-shell`

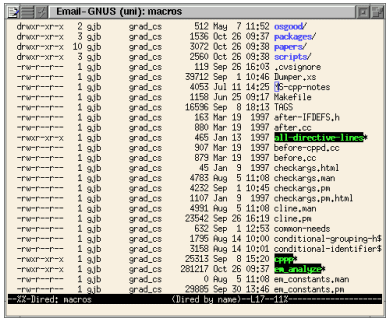
Tags

- Prepare TAGS file with (one of):
  `etags *.[ch]`
  `etags *.el`
- `M-.` or `M-x find-tag`
  Primed with identifier near point
  Completes
- `M-x tags-search`
- `M-x tags-query-replace`
- `M-x tags-apropos`
- `M-x visit-tags-table`
  Switch currently used TAGS file

Dired (directory editor)

- `M-x dired` or `C-x d` or find-file on a directory
- Omit uninteresting files
- Sort
- Operate on files
  - copy, rename, delete
  - print, chmod, (un)compress
  - `!` = shell command (with DWIM)
  - mark multiple files (with regexp)
- `g`: refresh

Dired screen shot



Omitting, sorting, DWIM with **!**

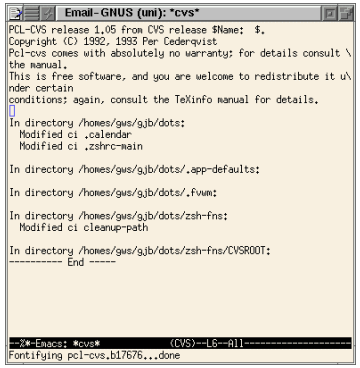
Ange-ftp, EFS

- Filenames in a special form:
  `/mernst@cf-web.mit.edu:/www/index.html`
- Edit from home, edit non-NFS-accessible files
- Use as a general ftp client
  `/anonymous@prep.ai.mit.edu:/pub`
  (works with Dired)

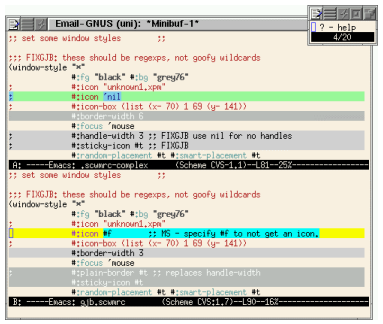
Version control

- RCS:
  `C-x C-q` vc-toggle-read-only
  `C-x v v` vc-next-action
- CVS:
  Above keystrokes, but also pcl-cvs mode for updating and managing module
  `M-x cvs-update`

pcl-cvs mode



Ediff mode



Crypt++

`(require 'crypt++)`

- Transparently handle:
  - compressed
  - archives (via tar-mode)
  - encrypted
  - newline variants (Unix vs DOS)
  - combinations of these
- Other options:
  jka-compr, dos-mode

Emacs server

- Reuse existing Emacs process (faster, less memory)
- Emacs: (server-start)
  To edit files: `emacsclient`
- XEmacs: (gnuserv-start)
  `gnuclient`
  To finish editing: `C-x #`
- Use `VISUAL`, `EDITOR` and `EDIT` environment variables

More about the server

- Emacs on other machine?
  Use `gnuserv` (even for Emacs)
  ```
  gnuclient -h EMACS_HOST -r \
    /$USER\@$HOSTNAME:
  ```
- `-r` option gives prefix to tell Emacs where how to get to the root filesystem of the machine you invoke `gnuclient` from
- Use a fallback:
  ```
  setenv EDITOR gnuclient-or-other
  ```

```
#!/bin/sh - # gnuclient-or-other
if [ "$DISPLAY" = "" ];  then jed "$@" else
  gnuclient "$@" || pico "$@"
fi
```


Amusements

- hanoi, life
- madlib, cryparithmetic puzzles adventure games
- tetris
- yow

Doctor

My advisor hates me.
Why do you say advisor hates you?

He criticizes my research, and his mother wears combat boots.
Tell me something about his family.

She's a bearded lady. Should I drop out of school to join the circus?
Tell me about your school.

My office is in a beautiful building.
What makes you think your office is in a beautiful building?

I think very little.
Possibly problems at school are caused by this.

Other packages

- calendar
- desktop
- complete library
  `M-X -rectangle<TAB>`
- two-column library