entr Run on file change

CONCEPTS

entr reads the file list from stdin, so it only watches what you pipe in. It never searches directories on its own.

File list ls, find, git ls-files
↓ stdin
entr watches those files
↓ change
Utility run or restart
↓ exit
Wait for the next change

Events that arrive while the command runs are merged into one by default, so saving twice mid-build gives one rerun. Add -a to react to all of them.

RUNNING

`<list> | entr <cmd> [args]`
Run a command whenever a listed file changes

`entr -r <cmd>`
Keep it running; restart on change

`entr -c <cmd>`
Clear the screen before each run

`entr -p <cmd>`
Postpone: skip the first run

`entr -s '<shell string>'`
Run a string through $SHELL (pipes, &&)

`entr -cc <cmd>`
Clear screen and scrollback

`entr -z <cmd>`
Exit after the command finishes

`entr -rz <cmd>`
Restart on change, exit when it dies

`entr -n <cmd>`
Non-interactive: leave the TTY alone

FEEDING THE LIST

`ls | entr <cmd>`
Watch the files in this directory

FEEDING THE LIST (cont.)

`ls *.py | entr <cmd>`
Watch files matching a glob

`find . -name '*.go' | entr <cmd>`
Watch a tree by pattern

`git ls-files | entr <cmd>`
Watch tracked files only

`fd -e rs | entr <cmd>`
Use fd for the list

`echo <file> | entr <cmd>`
Watch exactly one file

`find src tests | entr <cmd>`
Watch several directories

`git ls-files -co --exclude-standard | entr <cmd>`
Tracked and new unignored files

`find . -type f -not -path './.git/*' | entr <cmd>`
Skip a directory

CHANGED FILE AND DIRECTORIES

`/_`
Placeholder: path of the file that changed

`entr <cmd> /_`
Pass the changed file as an argument

`entr -s 'echo $0'`
In -s mode the changed file is $0

`entr -d <cmd>`
Watch parent dirs; exit 2 on new entry

`while sleep 1; do <list> | entr -d <cmd>; done`
Rescan the list after a new file

KEYS AND ENVIRONMENT

Space
Run the command now

q / Ctrl+C
Quit entr