Subject: Re: autocomplete

Posted by savoie on Fri, 01 Apr 2005 15:10:38 GMT

View Forum Message <> Reply to Message

maarten <user@domain.invalid> writes:

- > Thanks this is a major improvement!
- > Is there however another option for the 'arrow up' question, because I am not
- > really happy with M-r and having to repeat that many times.

Yeah, JD Smith implemented this for me a while ago.

```
put this in your .emacs to add up and down arrow functionality in the shell
(add-hook 'idlwave-shell-mode-hook
      (lambda ()
       ;; A cheat for quick function-only lookup in the shell.
       (local-set-key [up] 'jds-history-search-up)
   (local-set-key [down] 'jds-history-search-down)))
Then make sure these functions are available I put them into another file and
make sure that's loaded at start up before I add the hooks above.
;; Make arrows search on text between point and process-mark
(defvar jds-search-regexp nil)
(defun jds-search-pre-command-hook ()
 (unless (or (eq this-command 'jds-history-search-up)
    (eq this-command 'jds-history-search-down))
  (remove-hook 'pre-command-hook 'jds-search-pre-command-hook t)
  (setq jds-search-regexp nil)))
(defun jds-history-search (dir)
 (when (null ids-search-regexp)
  (let ((str (buffer-substring (point) (save-excursion (comint-bol nil)
     (point)))))
   (if (or (string= str "") (not (looking-at "\\s-*$")))
 (setq jds-search-regexp ".")
:: This is what JDS had.
;; (setq jds-search-regexp (concat "^" (regexp-quote str)))
(setq jds-search-regexp (regexp-quote str)))
   (add-hook 'pre-command-hook 'jds-search-pre-command-hook nil t)))
 (comint-previous-matching-input jds-search-regexp dir))
```

```
(defun jds-history-search-up ()
 (interactive)
 (jds-history-search 1))
(defun jds-history-search-down ()
 (interactive)
 (jds-history-search -1))
This should help.
Matt
```

Matthew Savoie - Scientific Programmer National Snow and Ice Data Center (303) 735-0785 http://nsidc.org