

---

Subject: IDL + Emacs' shell mode = an IDL 'next-error' elisp function

Posted by [alans](#) on Tue, 18 Aug 1992 20:49:35 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

When a colleague complained that there was no way to pop emacs to IDL's error line (like next-error in compile.el), I decided to try to write one.

If you run IDL from <M-x> shell in emacs, the function at the end of this article will do this, although it's not a masterpiece as elisp goes :)

I have some other questions about running IDL from emacs shell mode, though... Specifically, what the heck is IDL doing with the terminal? All my typed-in commands get echoed back, ^M's get written all over the place, keypad arrow keys don't work, from the IDL perspective (i.e., no command recall; you just move around the shell buffer) ... Clue: TERM is set to "emacs" when in shell mode. I assume IDL hasn't the slightest idea what to do with this and things go south when IDL tries to setup the keys :( I'm running IDL 2.3.2, Emacs 18.55, and I run Emacs from within an Xterm on a Sun 3/80, for what it's worth.

Any ideas, please post or email me...anyway, here's the elisp. Hope it helps somebody. If you improve it, please send me the improvements. Thanks!

(note the followup-to comp.lang.idl-pvwave if your newsreader is brain-dead :)

---cut here---

```
(defun idl-last-error ()
```

```
"
```

```
idl (via shell) equivalent of next-error; find file and jump to error line.
```

```
this actually does work, but should be fixed up to do regexp searches.
```

```
"
```

```
(interactive)
```

```
(let ((idl-here (point)))
```

```
  (search-backward "\n")
```

```
  (forward-char 1)
```

```
  (let ((idl-prompt (buffer-substring (point) idl-here)))
```

```
    (search-backward idl-prompt)
```

```
    (cond ((search-forward "ted at " idl-here t)
```

```
      (progn
```

```
        (search-forward "<")
```

```
        (let ((here (point)))
```

```
          (search-forward "(")
```

```
          (let ((file (buffer-substring here (- (point) 1)))
```

```
                (here (+ (point) 1)))
```

```
            (search-forward ")")
```

```
            (let ((line-no (buffer-substring here (- (point) 1))))
```

```
              (goto-char idl-here)
```

```
              (find-file-other-window file)
```

```
              (goto-line (string-to-int line-no))))))
```

```
            ((search-forward "At:" idl-here t)
```

```
              (progn
```

```

      (let ((here (+ (point) 1)))
(search-forward ",")
(let ((file (buffer-substring here (- (point) 1))))
  (search-forward "Line")
  (let ((here (point)))
    (search-forward "\n")
    (let ((line-no (buffer-substring here (- (point) 1))))
      (goto-char idl-here)
      (find-file-other-window file)
      (goto-line (string-to-int line-no))))))
(t
 (goto-char idl-here)
 (message "no error messages found"))))

```

```

(define-key shell-mode-map "\C-x`" 'idl-last-error)
---cut here---

```

```

--
      Alan J.Stein   MIT/Lincoln Laboratory   alans@ll.mit.edu

```

```

*****
*                                     *
*      On behalf of the Stein family (and the rest of Somerset, NJ),      *
*      congratulations to Olympic 10-meter platform diving Silver        *
*      medalist Scott Donie!                                           *
*                                     *
*****

```