NAME

pump − Shell data transfer command

SYNOPSIS

pump [−[subchar]] [+] [eofstr]

DESCRIPTION

Pump is a filter that copies its standard input to standard output with possible substitution of Shell arguments and variables. It reads its input to end-of-file, or until it finds*eofstr* alone on a line. If not specified, *eofstr* is assumed to be '!'. Normally, Shell variable and argument values are substituted in the data stream, using '$' as the character to indicate their presence. The argument '−' alone suppresses all substitution, '−*subchar*' causes *subchar* to be used as the indicator character for substitution in place of '$'. Escaping is handled as in double quoted(") strings: the indicator character may be hidden by preceding it with a '\'. Otherwise, '\' and other characters are transmitted unchanged. The '+' flag causes all leading tab characters in the input to be thrown away, in order to permit readable indentation of text and *eofstr.* *Pump*may be used interacti vely and in pipelines. A common use is to get variable values into editor scripts. If $a, $b, and $c have the values A, B, and C respectively, the two sequences below are equivalent:

```
pump −~ | ed file          ed file
1,$s/~a$/~b/               1,$s/A$/B/
?~c?                       ?C?
!                          q
```

The sequence above will work at the terminal as well as in Shell procedures. *Pump* is an ef ficient and convenient replacement for multiple uses of *echo(I);* e.g., the following are equivalent:

```
pump >file                 echo "$1" >file
$1                         echo "$2" >>file
$2
!
```

Pump is actually implemented inside the Shell, although it executes as a separate process.

SEE ALSO

echo(I), sh(I)

BUGS

The size of *eofstr* is limited to 95 bytes, and it may not begin with '+'.