

Corso di Fondamenti di Informatica 2
CdL Ingegneria Informatica
Ing. Franco Zambonelli

IL SISTEMA OPERATIVO UNIX: L'EDITOR VI

Lucidi Realizzati in Collaborazione con:

Prof. Letizia Leonardi
Università di Modena

Prof. Antonio Corradi
Università di Bologna

Prof. Cesare Stefanelli
Università di Ferrara

EDITOR STANDARD DI UNIX: VI

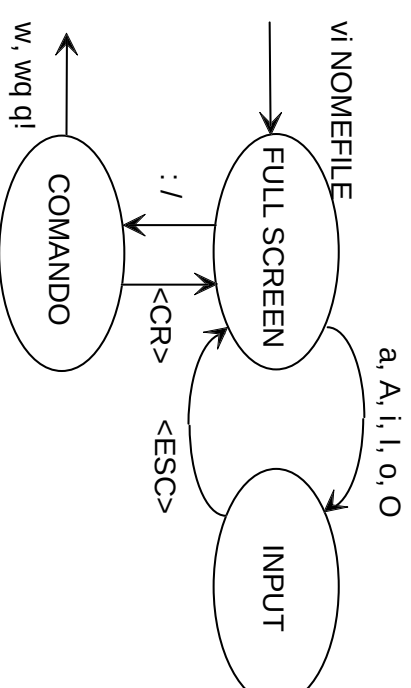
vi (**V**isual display editor)

- è l'editor standard di Unix
 - non basato sul concetto di WINDOW (finestra)
 - ma usabile da un qualunque tipo di terminale
- È un Editor **a linee**, **non full-screen**

Vi ragiona a stati:

- 1) lo stato comandi display (*full screen*)
- 2) lo stato riga comandi (*comando*)
- 3) lo stato di input (*input*)

Diagramma a stati di VI:



legenda: Maiuscole prima del cursore, minuscole dopo il cursore

Per invocarlo: **vi <nomefile>**

il file di nome <nomefile> viene aperto se esiste, o creato altrimenti. Ha inizio l'editing e l'accettazione dei comandi.

CASO NUOVO FILE → vi prova.c

<=== cursore

~

~

~

~

~

~

~

"prova.c" [New file]

<===

RIGA

COMANDI

CASO FILE ESISTENTE → vi prova.c

#include <stdio.h>

main(argc, argv)

int argc;

char ** argv;

{int i;

for (i = 0; i < argc; i++)

printf("argv[%d] = %s\n", argv[i]);

}

~

"prova.c" 9 lines, 138 characters

<===

RIGA

COMANDI

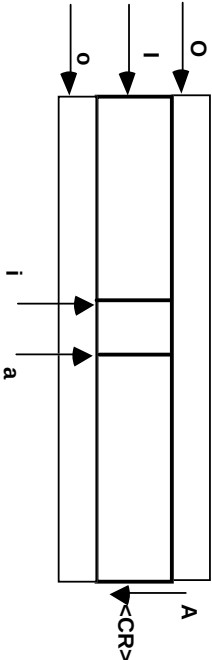
Nel caso di file lungo, viene visualizzata la parte iniziale

Il file è considerato composto da un insieme di linee, ciascuna completata dal terminatore di linea (<CR>)

Ogni linea è formata da parole e da caratteri, e ci sono comandi adatti per ogni possibile azione.

COMANDI di INPUT

Si passa da stato comandi a stato input nelle diverse posizioni :



COMANDI DI SCHERMO

MOVIMENTO

di un carattere

← h ↓ j → k → i

<blank> carattere successivo

<backspace>carattere precedente

di una parola

w (dopo), b (prima)

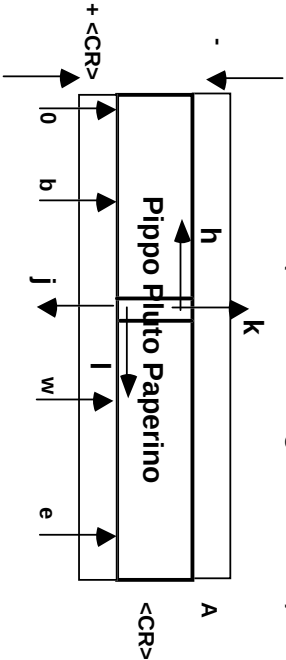
e (fine della parola corrente)

di una riga

<CR> o + (prossima)

- (inizio della riga precedente)

\$ (fine della riga corrente)



Alcuni comandi possono essere preceduti da un numero che indica **quante volte** il comando deve essere **ripetuto**

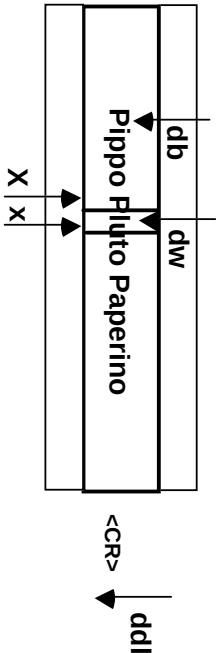
Ad esempio: **3w** sposta il cursore in avanti di 3 parole

VISUALIZZAZIONE

visualizza videata seguente	<CTRL> F (indicato ^F)
visualizza videata precedente	<CTRL> B (indicato ^B)
rinfresca il buffer (corrente)	<CTRL> L (indicato ^L)

CANCELLAZIONE

di un carattere	x (corrente), X (prima)
di una parola	dw (dopo), db (prima)
di una riga	dd
di tutto il resto della riga corrente	d\$ o D



SOSTITUZIONE

di un carattere	r <ch>
di un testo al posto del carattere	R testo <ESC>
di un testo al posto della parola corrente	cw testo <ESC>

AZIONI VARIE

ripete l'ultima azione	.
annulla l'ultima azione eseguita (undo)	u
annulla tutte le azioni sulla linea	U
unisce la linea corrente con la seguente	J
ripete la ricerca in avanti	n
ripete la ricerca all'indietro	N

COMANDI con uso di BUFFERIZZAZIONE

Esiste un buffer corrente senza nome, in cui sono anche memorizzate le ultime parti distrutte

copia la riga corrente in un buffer	nY nyy
copia il buffer sopra la riga corrente	P
copia il buffer dopo la riga corrente	p
(maiuscolo prima, minuscolo dopo)	

altri buffer (nomi da a a z , p.e. f)	"fnyy
salvataggio e ripristino	"fnp "fnp

Esempi:

a) COPIA DI UN INSIEME DI RIGHE

- posizionare il cursore sulla prima riga da copiare
- **<n>Y** copia **<n>** righe nel buffer
- spostare il cursore sulla riga sotto la quale copiare
- **p** inserisce le righe prelevate dal buffer

b) SPOSTAMENTO DI UN INSIEME DI RIGHE

- posizionare il cursore sulla prima riga da spostare
- **<n>dd** muove **<n>** righe nel buffer
- spostare il cursore sulla riga sotto la quale copiare
- **p** inserisce le righe prelevate dal buffer

COMANDI ULTERIORI

COMANDI dalla RIGA di COMANDO

RICERCA

di una stringa specificata (avanti) /stringa
di una stringa (indietro) ?stringa

MOVIMENTO del cursore

su una linea specificata :numerolinea
ultima linea del file :\$

ELIMINAZIONE

di linee del file :n,md linee da n a m

INTERAZIONE con il FILE SYSTEM

lettura ed inserimento di file alla posizione corrente
:r nomefile

lettura ed inserimento di file dopo la linea n
:nr nomefile

uscita da vi :q
senza salvare :q!

scrittura su file :w nomefile
parti su file :n,mw nomefile
aggiunte su file :n,mw >> nomefile
forzata su file :w! nomefile

superando i normali controlli (se possibile)
specifica di opzioni :set opzione
visualizza i numeri di riga :set number
elimina i numeri di riga :set nonumber

ESECUZIONE di un qualunque comando di SHELL
escape ad uno shell :!
comando di shell>

Esempi di comandi:

a) SALVA ED ESCE DALL'EDITOR

- <ESC> esce dallo stato input
- :wq salva ed esce dall'editor

ATTENZIONE:

Se il file non consente la scrittura da parte dell'utente
che sta usando il vi, l'editor riporterà un errore quando
si tenta di scrivere sul file

b) ESCE DALL'EDITOR SENZA SALVARE

- <ESC> esce dallo stato input
- :q! esce dall'editor senza salvare

ATTENZIONE:

Se si usa :q senza aver salvato il file, l'editor avvisa
No write since last change (:q!ti overrides)

Esistono moltissimi altri comandi: nonostante la scarsa
leggibilità, vi è l'unico editor presente in tutti i sistemi
UNIX.

Si consiglia di approfondirne la conoscenza.