

Funcionamiento general del vi

Al invocar este editor aparece en el monitor la pantalla de edición. En ella aparece la posición del cursor resaltada, las líneas en blanco señaladas con el carácter ~ y en la parte inferior de la pantalla aparece la línea de estado, que muestra el nombre del fichero y el número de caracteres que contiene.

Si se invoca el vi pasándole como parámetro el nombre de un fichero en la pantalla de edición aparecerá su contenido. Cuando se invoca este editor con el nombre de un fichero que no existe, se crea automáticamente.

Existen dos modos de operación en el vi :

Modo Edición:

Para añadir texto al fichero

Modo Comando:

Para introducir órdenes que realizan funciones específicas del vi .

Cuando se edita un fichero con el vi , los cambios no se hacen directamente sobre el fichero. En realidad, se aplican a una copia del fichero que el vi crea en un espacio de memoria temporal llamado buffer. La copia en disco del fichero se modifica sólo cuando se graban los contenidos del buffer.

Esto tiene sus ventajas y sus inconvenientes. Por un lado, significa que se puede salir de la edición de un fichero y descartar todos los cambios hechos durante una sesión, dejando la copia de disco intacta. Por otro lado, se podría perder el contenido no grabado del buffer de trabajo si el sistema cae.

Lo más aconsejable es grabar el trabajo frecuentemente, especialmente cuando se hacen cambios importantes. Para grabar el fichero que se está editando sin salir del vi , basta pulsar en modo comando la orden :w

Para terminar la sesión caben varias posibilidades, siempre en modo comando:

:q
Salir cuando no se han hecho modificaciones

:q!
Salir y descartar los cambios

:wq
Salir y guardar los cambios

IMPORTANTE: Hay que procurar no editar un fichero con varias copias del vi al mismo tiempo, pues de este modo no se está seguro de cuál es la copia que está realmente grabada en disco.

Modo Edición:

Cuando se arranca el `vi`, siempre está en modo comando, por lo que antes de poder escribir texto en el fichero se debe teclear uno de los comandos de entrada del `vi`, tales como `i` (insert), para insertar texto en la posición actual del cursor, o `a` (append) para insertar texto después de la posición actual del cursor.

Para regresar la modo comando, basta con presionar Esc. Si en un momento determinado no se sabe en qué modo se está, simplemente pulsando Esc se asegura uno de que está en modo comando, y se podrá continuar con el trabajo.

Modo Comando:

Cuando se abre un fichero con `vi`, se está en modo comando. En este modo se pueden introducir comandos que implementan un amplio rango de funciones. Muchos de estos comandos constan de una o dos letras y un número opcional relacionado con distintas funciones.

Muchos comandos no requieren pulsar Return para que se ejecuten, pero todas las órdenes que empiezan por dos puntos (:) requieren pulsar Return para que se ejecuten.

Moverse por un fichero

Cuando arrancamos el `vi`, el cursor esta en la esquina superior izquierda de la pantalla. En modo comando, existen órdenes que nos permiten moverlo por toda la pantalla.

1. De carácter en carácter

- Si el teclado esta provisto con las flechas, se pueden utilizar para mover el cursor libremente por el texto editado hasta el momento.
- Si se usa el `vi` desde un terminal remoto, las flechas pueden no funcionar correctamente, depende del emulador de terminal. En dicho caso las teclas usadas son:

Izquierda:

h ó SpcBar

Derecha:

l ó BckSpc

Arriba:

k

Abajo:

j

2. De palabra en palabra

- Pulsando `w` (word) nos movemos una palabra hacia la derecha.
- Pulsando `b` (before) nos movemos una palabra hacia la izquierda.

3. Dentro de una línea

- Pulsando `^` nos movemos al comienzo de la línea en la que está el cursor.

- Pulsando \$ nos movemos al final de la línea actual.
- Pulsando Return nos movemos al comienzo de la línea siguiente.

4. Dentro de una pantalla

- Pulsando H (high) nos movemos a la parte superior de la pantalla.
- Pulsando L (low) nos movemos a la parte inferior de la pantalla.
- Pulsando M (middle) nos movemos a la mitad de la pantalla

5. Av/Re Página

- Pulsando Ctrl-F (forward) se avanza una pantalla, moviéndose el cursor a la esquina superior izquierda de la nueva pantalla.
- Pulsando Ctrl-D se avanza media pantalla.
- Pulsando Ctrl-B (scroll backward) se retrocede una pantalla.
- Pulsando Ctrl-U se retrocede media pantalla.

Insertar texto

vi proporciona muchos comandos para insertar texto, que nos hacen pasar del modo comando al modo de edición.

1. Añadir

- Pulsando a (append) se inserta texto a la derecha del cursor.
- Pulsando A se añade texto al final de la línea en la que está el cursor.

2. Insertar

- Pulsando i (insert) se inserta texto a la izquierda del cursor.
- Pulsando I se inserta texto al principio de una línea.

3. Insertar una línea

- Pulsando o (open) se inserta una línea debajo de la posición actual del cursor
- Pulsando O se inserta una línea encima de la actual posición del cursor.

Cambiar texto

Cambiar texto implica sustituir una sección de texto por otra. *vi* tiene varios modos de hacer esto, dependiendo de lo que se desee cambiar.

1. Carácter por Carácter/es:

Substituir caracteres

Para substituir el carácter bajo el cursor por uno o mas caracteres, teclear s seguido del nuevo texto y pulsar Esc cuando se acabe.

Reemplazar un carácter

Para reemplazar el carácter bajo el cursor por otro carácter, pulsar r, seguido por un unico carácter, pues al pulsar una tecla el *vi* inmediatamente regresa a modo comando

Trasposicionar caracteres

Esto es útil para fallos tales como escribir *geu* en lugar de *que*. Colocando el cursor sobre la primera letra a mover y pulsando *x*, se intercambian las posiciones de ambos caracteres.

2. Palabra por palabra/s

Palabra completa

Posicionar el cursor al principio de la palabra a ser reemplazada, teclear *cw* seguido de la nueva palabra. Cuando se termine de modificar se pulsará *Esc*

Parte de una palabra

Colocar el cursor sobre la palabra, a la derecha de la parte a cambiar, y proceder como en el caso anterior.

3. Línea por línea/s

Línea completa

Poner el cursor en cualquier parte de la línea y teclear *cc*. La línea desaparece, dejando una línea en blanco para el nuevo texto, que puede ser de cualquier longitud. Para terminar se pulsará *Esc*.

Parte de una línea

Colocar el cursor a la derecha de la parte a modificar. Pulsar *C*, introducir la corrección y pulsar *Esc* para terminar.

4. Partir/Unir líneas

Partir una línea

Mover el cursor al espacio donde se quiere partir la línea y teclear *r* (*replace*) seguido de *Return* (se reemplazaría el espacio por un *return* sin afectar al texto)

Unir dos líneas

Colocar el cursor en la línea superior y teclear *J*.

Deshacer Cambios

vi proporciona dos modos para deshacer los cambios hechos mientras no se ha grabado el fichero a disco.

- **Deshacer el comando previo.**
 - Pulsando *u* a continuación del último comando, se deshacen los cambios producidos por la ejecución del mismo.
- **Deshacer los cambios de una línea.**
 - Pulsando *U* se deshacen todos los cambios que se han hecho sobre una línea. Este comando funciona sólo si se permanece en dicha línea.

Borrar texto

1. Caracteres

- Para borrar un carácter, posicionar el cursor sobre el carácter a borrar y teclear *x*. El comando *x* también borra el espacio ocupado por el carácter.
- Para borrar el carácter anterior a la posición del cursor pulsar *X*.

2. Palabras

- Para borrar una palabra, posicionar el cursor al principio de la palabra y pulsar dw, entonces se borrara la palabra y el espacio que ésta ocupaba.
- Para borrar parte de una palabra, hay que colocar el cursor a la derecha de la parte a modificar, y teclear dw.

3. Lineas

- Para borrar una línea y el espacio que ésta ocupa, basta con pulsar dd.
- Para borrar todo lo que este a la derecha del cursor, basta con pulsar D.
- Para borrar todo lo que este a la izquierda del mismo basta con pulsar d0.

4. Fichero

- Para borrar desde la línea actual hasta el final del fichero, basta con teclear dG.
- Para borrar desde el principio del fichero hasta la línea actual, basta teclear d1G.

Copiar y mover texto

De la misma manera que muchos procesadores de texto permiten copiar/pegar, o cortar/pegar líneas de texto, el vi incluye también esta posibilidad, mediante los comandos *yank /put* y *delete /put* , respectivamente.

1. Copiar y pegar

El procedimiento para copiar una ó más líneas, es el siguiente:

- Colocarse en la primera línea a copiar.
- Escribir el número de líneas que se desea copiar.
- Teclear yy ó Y (comando *yank*).
- Colocarse en la línea a donde se desean copiar
- Teclear p ó P (comando *pegar*)

2. Cortar y pegar

El procedimiento para mover una ó varias líneas es el siguiente:

- Colocarse en la primera línea a cortar.
- Escribir el número de líneas que se desea cortar.
- Teclear dd (comando *delete*).
- Colocarse en la línea a donde se desean pegar
- Teclear p ó P (comando *pegar*)

3. Usar buffers con nombre

Para insertar repetidamente un grupo de líneas en varios lugares dentro de un documento, se pueden copiar las líneas a un buffer con nombre. El procedimiento para dar nombre a un buffer es el siguiente:

- Colocarse en la primera línea a copiar.

- Teclear comillas dobles (") seguido de la letra con que se desea nombrar el buffer (Ej: "a)
- A continuación escribir el número de líneas que se desea copiar o cortar.
- Teclear yy ó Y (comando *yank*) ó dd (comando *delete*).
- Colocarse en la línea donde se desean pegar.
- Teclear comillas dobles seguidas por la letra del buffer cuyo contenido se desea insertar y por p ó P (comando *pegar*).

Usar los comandos del ex

Cuando se trata de grandes bloques de texto, los comandos del *ex* son más seguros que *yank* , *delete* y *put* del *vi*.

Éstes permiten especificar el rango de líneas a mover o copiar, así como la línea anterior al punto donde se insertarán las líneas.

Para conocer el rango de líneas, es necesario conocer el número de cada línea:

Para visualizar los números de línea

```
:set nu
```

Para ocultar los números de línea

```
:set nonu
```

Conocidos los números de las líneas que se van a copiar, las órdenes del editor *ex* disponibles son:

1. Copiar líneas

El formato básico del comando para copiar en el *ex* es el siguiente:

```
:#línea,#línea co #línea
```

2. Mover líneas

El formato básico del comando para mover en el *ex* es el siguiente:

```
:#línea,#línea m #línea
```

3. Borrar líneas

El formato básico del comando para borrar en el *ex* es el siguiente:

```
:#línea,#línea d
```

En todos los casos, los dos primeros números de línea especifican el rango de líneas a copiar , cortar o borrar, respectivamente, y el último corresponde a la línea anterior al punto de inserción.

Se pueden utilizar abreviaturas para especificar los rangos de líneas:

- Denota la línea actual
- \$ Denota el final del fichero

Buscar y reemplazar

`vi` proporciona varios modos de encontrar un string de caracteres en el texto, así como de reemplazarlo.

Un string es simplemente uno o más caracteres en fila. Puede incluir letras, números, signos de puntuación, caracteres especiales, espacios en blanco, tabulaciones o retornos de carro. Un string puede ser una palabra gramatical o puede ser una parte de una palabra.

Para ir a una línea determinada de un fichero abierto se teclearía:

`#líneaG` (Ej. `6G`)

Si no se indica un número de línea se consideraría la última del fichero.

Búsqueda

Para encontrar un string, hay que escribir una barra hacia la derecha (`/`) seguida por el string que se desea buscar, y pulsar `return`.

El `vi` posiciona el cursor en la siguiente ocurrencia del string. Tecleando `n` se va a la siguiente ocurrencia del string, y pulsando `N` a la anterior.

Si en lugar de una barra se escribiese una interrogación (`?`), la búsqueda sería hacia atrás, y las direcciones de las órdenes `n` y `N` se invertirían.

Normalmente las búsquedas son *case-sensitive*. Si se quiere que `vi` ignore mayúsculas y minúsculas durante la búsqueda, hay que teclear previamente:

`:set ic`

Para volver al cambio por defecto se teclearía:

`:set noic`

Ciertos caracteres especiales (`/ & ! . ^ * $ \ ?`) tienen un significado particular en el proceso de búsqueda, y por lo tanto deben especificarse de un modo especial cuando están incluidos en el string a buscar, precediéndolos por la barra hacia atrás (`\`).

Refinando la búsqueda

Se pueden realizar búsquedas más precisas, marcando el string con indicadores de características tales como:

- **Principio de línea**

Para intentar emparejar el comienzo de una línea, hay que escribir antes del string a buscar el carácter `^`

- **Fin de línea**

Para emparejar el final de una línea, hay que terminar el string a buscar por el carácter `$`

- **Principio de palabra**

Para emparejar el principio de una palabra, hay que teclear \<

- **Fin de palabra**

Para emparejar el final de una palabra, hay que teclear \> al final del string.

- **Comodines**

- Para emparejar cualquier carácter, hay que teclear un punto en el string en la posición a emparejar. (igual que ? en los nombres de ficheros de MS-DOS)
- Para buscar caracteres alternativos en un string, se deben poner las alternativas entre corchetes, por ejemplo:
 - [md]xxx encontraría los strings mxxx y dxxx
 - [d-m]xxx encontraría todos los strings que comiencen por cualquier letra de la d a la m y sigan con xxx.
- Para emparejar cero ó más ocurrencias del último carácter escrito, basta añadir un * en el string.

Reemplazo

El procedimiento para reemplazar un string está es similar al de búsqueda arriba mencionado. Todos los caracteres especiales de emparejamiento se pueden usar también en una búsqueda+reemplazamiento.

La forma básica de la orden es:

```
:g/string_buscado/s//string_cambiado/g
```

Se puede modificar el comando para parar la búsqueda y hacer que el vi pida confirmación antes de realizar el reemplazo, mediante la orden gc (consult), respondiendo y (sí) o n (no).

Para terminar el proceso bastaría con pulsar Ctrl-C

Insertar un fichero en otro

vi permite insertar un fichero dentro del que se está editando, mediante la orden:

```
:#línea r nombre_fichero
```

Si no se especifica un número de línea, se tomará la posición actual del cursor