

TERMINALES MODO TEXTO EN CLIP

SACL a882sacl@yahoo.com.ar

Grupo Clip clip-castellano@gruposyadoo.com.ar

Versión 0.1 - 12/06/2005

Revisiones:

0.3 – 26/06/2005

INTRODUCCIÓN

Cuando vamos a desarrollar una aplicación Clip, uno de los elementos más importantes es la interfaz de usuario, lo que va a apreciar el usuario por pantalla. En el caso del mundo DOS, estábamos acostumbrados a verlo en una representación de “caracteres” por filas y columnas, ya sea de 25*80 ó 50*80.

En una primera etapa, supondremos que nuestras aplicaciones “texto” Clipper serán portadas a “texto” Clip. Lo normal en linux, por defecto, es que las terminales virtuales vengan configuradas como 25 filas por 80 columnas y en modo emulación “linux”.

Además debemos mantener en mente, que nuestra idea es que algunos equipos Windows (que nunca faltan) se conecten a un Servidor Central, por lo que deberán usar un “cliente” para esa conexión. Aquí viene el detalle, debemos estar conscientes de los modos de configuración de éste cliente para que esa relación sea compatible.

Por ejemplo, existe un cliente para conexión remota segura (para Windows) llamado Putty. Este simula una terminal grafica de linux (xterminal), que se puede configurar para emulación modo “linux” (entre otras) y variable número de filas y columnas.

Al portar o programar en Clip, debemos tener en cuenta esta potencial variabilidad de interfaz de salida. Por eso se recomienda encarecidamente, evitar el uso de posicionamiento fijo y recurrir a las funciones de posicionamiento relativo: maxrow() y maxcol().

Si aún después de todas estas recomendaciones, aún tenemos ganas de cambiar la apariencia de nuestra terminal de texto linux, entonces lean lo que viene a continuación.

TIPS PARA CAMBIAR RESOLUCION

(1) Usando nuestro cargador de arranque.

LILO: Editamos el archivo “/etc/lilo.conf” y agregamos o modificamos:

vga=modo

GRUB: Editamos el archivo “/boot/grub/menu.lst” y agregamos o modificamos:

```
title      Debian GNU/Linux, kernel 2.4.27-2-386
root       (hd0,5)
kernel     /boot/vmlinuz-2.4.27-2-386 root=/dev/hda6 vga=modo ro
initrd     /boot/initrd.img-2.4.27-2-386
savedefault
boot
```

Estas características permiten el uso de varios modos de video especiales soportados por el video de la BIOS. Debido a esto, la selección está limitada al tiempo de boot y trabaja sólo sobre máquinas 386.

Aquí en el fondo, lo que estamos haciendo es ocupar una área de memoria RAM para uso de VIDEO, algo como convertir nuestra terminal de texto en gráfica. Tanto es así, que en estas condiciones hasta podemos colocar imágenes o ver películas en este inhóspito ambiente.

A esta capacidad, de alto nivel, se le denomina “frame buffer”. Existen algunas tablas para que se guíen en lo que deben colocar en “**modo**” (argumento de VGA). No olvidar que esta capacidad debe estar incluida en nuestro kernel, nosotros lo único que hacemos es “activarla”.

Modos (como constantes para filas y columnas):

```
normal      80x25
extended ó ext 80x50
ask         Menú de selección.
```

Modos (para filas y columnas):

```
0x0f00      estándar 80x25
0x0f01      estándar con font de 8 puntos: 80x43 sobre EGA, 80x50 sobre VGA
0x0f02      VGA 80x43 (VGA switchado a 350 líneas “scan” con font de 8 puntos)
0x0f03      VGA 80x28 (estándar VGA “scans”, pero con font de 14 puntos)
0x0f04      Deja el corriente modo de video
0x0f05      VGA 80x30 (480 “scans”, 16-puntos font)
0x0f06      VGA 80x34 (480 “scans”, 14-puntos font)
0x0f07      VGA 80x60 (480 “scans”, 8-puntos font)
0x0f08      Modificar gráficos
```

Si por alguna razón LILO no soporta el modo en forma hexadecimal, se lo pueden dar en formato decimal. Sería del 3840 al 3848.

Todas estas configuraciones dependen mucho del hardware que tengan. Yo las probé en 3 equipos, y en solo 1 me resultó. En los otros 2 cambiaba la resolución al principio pero luego se reseteaba a normal.

El sistema que me resultó en todos, es el que se basa en cambiar la resolución de acuerdo a la siguiente tabla:

HEXA	640x480	800x600	1024x768	1280x1024
256	0x301	0x303	0x305	0x307
32k	0x310	0x313	0x316	0x319
64k	0x311	0x314	0x317	0x31A
16M	0x312	0x315	0x318	0x31B

DECI	640x480	800x600	1024x768	1280x1024
256	769	771	773	775
32k	784	787	790	793
64k	785	788	791	794
16M	786	789	792	795

En un monitor de 15 pulgadas, la resolución de 640x480 equivale a 80x30 y la de 800x600 a 80x37 columnas x filas. Ustedes practiquen todo lo que quieran. Si han bajado los fuentes del núcleo, echen una mirada al archivo “/usr/src/linux/Documentation/svgatextmode.txt”.

(2) Usando una emulación por software.

resizecons : Haría lo mismo que “resize”, no lo probé.

resize : Lo único que hace es modificar las variables de ambiente COLUMNS y LINES. En la práctica no es muy útil.

svgatextmode : Este si que es de verdad. En debian está en los repositorios (al menos de Sarge), así que es llegar y bajarlo. Posee un archivo de configuración “/etc/TextConfig” bien autoexplicativo. No es llegar y ejecutarlo desde consola, primero se deben indagar los valores posibles viendo su archivo de configuración. Por ejemplo:

```
# SVGATextMode 80x34x9
```

Probé varios modos, y todos me funcionaron. Lo bueno es que posee una variedad muy grande de configuración, incluyendo los tipos y tamaños de fonts:

Filas : 25, 28, 29, 30, 32, 33, 34, 40, 43, 50, 60
Columnas : 40, 80, 100, 116, 132

console-tools : este paquete es un clon de kbd y contiene varias herramientas. La más útil y sencilla, es para cambiar el font de las letras del terminal. Se debe modificar el archivo “/etc/console-tools/config” con lo siguiente:

```
SCREEN_FONT=<fuente>
```

Donde “fuente” es el nombre del archivo sin la extensión, ubicado en “/usr/share/consolefonts”. Generalmente son archivos *.psf.gz.

LA VARIABLE TERM

Para linux en general, existe la variable de entorno denominada TERM . Linux sabe que tipo de terminal estamos utilizando, o mejor dicho, que tipo de "emulación de terminal" estamos usando (si usamos nuestras consolas virtuales), así los programas toman el valor de esta variable para ver en la base de datos "terminfo", cuales son las capacidades de la terminal en cuestión.

Si digitamos.”echo \$TERM” en cualquiera de las terminales virtuales, veremos que aparece la palabra "linux", ya que es el tipo de emulación usado por defecto por los Pcs. Muy relacionadas están también

las variables LINES y COLUMNS, cuyo significado es bastante obvio.

TERMCAP Y TERMINFO

No todas las terminales tienen las mismas capacidades, es decir, no todas manejan los mismos caracteres de escape y demases, si el sistema operativo (SO) envía un caracter o secuencia que un terminal no reconoce, pues obviamente éste no va a hacer nada, y tal vez en el mejor de los casos, es posible que algo se desconfigure en la pantalla (¿?).

Este tipo de situación fue muy común, aún después de haber realizado una serie de estandarizaciones ANSI e ISO.

Para hacer frente a este problema de estándares, se creó una base de datos llamada "termcap", es decir "Terminal Capabilities" (Capacidades de un Terminal), la cual fue sucedida por otra base de datos, más nueva, llamada "terminfo" (Información de Terminales).

Estas bases de datos se encuentran en algunos archivos del sistema, por ejemplo en Debian Sarge están ubicadas en "/etc/terminfo/*" ó "/usr/lib/terminfo/*", que es un directorio donde están ordenados por abecedario las rutinas de bajo nivel sobre definiciones de los principales tipos de terminales y cuenta con una sección para cada tipo de terminal que se esté usando. Así el sistema sabe de que es capaz el terminal y no envía señales equívocas. Los programas de aplicación utilizan esta base de datos, mediante el llamado a ciertas librerías C. Para más información, en un terminal digitar "man terminfo".

CONFIGURANDO TERMCAP PARA CLIP

En el caso de Clip, este usa la base de datos "terminfo". Una copia del sistema se instala en el directorio \$CLIPROOT/etc/terminfo, al momento de la compilación e instalación de Clip. Si defines la variable de ambiente TERMCAP en tu archivo de configuración ".bash_profile" o similar, se pueden usar algunas de las definiciones del archivo "\$CLIPROOT/etc/termcap" en ambiente de texto. Por ejemplo:

trueansi	ansi xenix	cons25r	xenix8	AT386 at386	AT386c at386c
xterm	xterm-std	xterms	vt100	vt200	vt340
pc3 ibmpc3	pc3a ibmpc3a	pc3nc	console linux linux-koi8 linux-c linux-koi8r fslinux		
con80x25	con80x28	con80x43	con80x50	con100x37	con100x40
con132x43	ct120				

Cada una de éstas etiquetas viene seguida por una configuración que hace uso de la base de datos (a bajo nivel) "terminfo". Para una descripción detallada de todas las opciones, ver "man termcap" o <http://www.linuxinfo.com/english/man5/termcap.html>.

Todos los nombres de los "terminales avanzados" tienen que estar registrados en la base de datos "/etc/terminfo". Las tablas-unicódigo estándar de linux localizadas en "/usr/share/consoletrans" son usadas para la traducción de las salidas del terminal.

Las tablas-unicódigo necesarias, han sido desempacadas y puestas en el directorio \$CLIPROOT/charsets. La distribución de Clip viene con algunas tablas preexistentes. Por el momento NO seteen la variable TERMCAP, más adelante en el apartado del "teclado" veremos si nos es de mayor utilidad.