

Tito's Telnet 1.11

Emulador Sencillo de Terminal para Telnet

MANUAL DE USUARIO

Por: Tito Hinostroza

Modificado: 21/12/2012

NOTAS SOBRE EL TITO'S TELNET

Tito's Telnet, es un programa libre. No se restringes los derechos de copiarlo, reproducirlo o distribuirlo, o usarlo de acuerdo a su criterio. Solo se pide que se incluya una referencia al autor: Tito Hinostroza

Desarrollé Tito's Telnet como una necesidad para desarrollar mi trabajo diario. No está pensado para ser un cliente completo que cumpla con todas las especificaciones del protocolo Telnet.

Solo incluye algunas implementaciones comunes del protocolo Telnet. No soporta ssh. Solo he implementado soporte a las principales secuencias de control del protocolo. No he implementado el reconocimiento del color. Sin embargo el Programa tiene una característica interna que colorea las palabras que van llegando por el puerto.

Está diseñado para poder trabajar con el Shell "Korn", así que puede comportarse algo limitado al trabajar en "bash" o "csh". Solo implementa un terminal en modo VT100 de 80 columnas por 25 filas.

Este programa no está para nada terminado (tampoco este documento). Se trata de una versión Beta, muy Beta. Falta corregir varios "bugs" y agregar nuevas funcionalidades.

A pesar de su simplicidad, ese programa soporta automatización mediante un lenguaje propio que es muy parecido al lenguaje de programación del TeraTerm. Además incluye un editor con resaltado de sintaxis y ayuda contextual muy útil.

Desgraciadamente no lo he podido revisar mucho, en los últimos dos años. Así que está un poco descuidado.

Van mis agradecimientos a:

- Jimmy Hung <http://www.planet-source-code.com/vb/scripts/ShowCode.asp?txtCodeId=72506&lngWId=1> , por su código sobre protocolo de Telnet.
- Flavio González Vázquez flavio@ya.com, por su valioso código para control del mouse.

¿POR QUÉ UN CLIENTE MÁS?

Habiendo ya muchos clientes telnet, la pregunta sería por qué crear uno más. Pues la respuesta es también común: Porque ninguno tenía las características que estaba buscando en un cliente de Telnet (en el año 2010). ¿Cuáles eran?

- Que incluyera una editor de texto independiente, para escribir los comandos. Este editor de texto debería ser independiente del “prompt” del terminal, que tiene opciones limitadas de edición. Además el editor debería tener opción, deshacer, resaltado de sintaxis, menús contextual, etc. Y debería permitir enviar todo un script completo al shell.
- El editor de texto independiente debería poder mostrarse siempre visible al lado del terminal (o abajo) y poder separarse como una ventana independiente.
- Debe poder cambiarse rápidamente con “Ctrl+Tab”, entre la ventana del editor y la ventana del terminal para poder ingresar los comandos a través del editor o directamente al terminal.
- El terminal (así como el editor independiente) debería tener resaltado de sintaxis en colores para el texto mostrado. Este coloreado del texto debería funcionar aún cuando el servidor remoto no envíe códigos de color (De hecho no es necesario implementar el reconocimiento del color en el cliente Telnet). Debería también mecanismos para pintar el “prompt” mediante algún mecanismo que le permita identificarlo en todo el flujo de texto recibido.
- La pantalla del terminal, debería tener opciones de exploración, como si se tratara de editor normal. Debería poder soportar las opciones de búsqueda, seleccionar, copiar y pegar texto en el “prompt”, usando el teclado o el Mouse. Se debería poder mantener la ventana fija aunque lleguen caracteres al terminal.
- El terminal debería poder redimensionarse libremente, e independientemente del tamaño “virtual” del terminal Telnet. Para ellos debería manejar barras de desplazamiento.
- El terminal debería ser rápido, sin retrasos en la visualización, ni parpadeos. Además debe permitir el registro del texto recibido.
- El cliente Telnet debería soportar automatización, a través de un lenguaje sencillo de macros, con un editor avanzado, con resaltado de sintaxis y ayuda contextual.
- Que sea portable. Sin necesidad de instalación.

Estos fueron los principales requerimientos que buscaba en un cliente de Telnet. Si existía uno que cumpliera todos estos requerimientos en el 2010, entonces mi trabajo ha sido en vano. Si no, entonces ha valido la pena.

INSTALACIÓN

El programa se diseñó para no requerir instalación. Solo basta con copiar y descomprimir la carpeta que contiene los archivos del programa “\TitosTelnet_1.11”.

El ejecutable es “TTelnet.exe”. Bastará con ejecutar este archivo para iniciar el programa. Si se obtiene algún mensaje de error con respecto al archivo “MSCOMM32.OCX”, se debe proceder a registrarlo en Windows.

Para registrar este “OCX”, se debe abrir una ventana de comandos “CMD”, y ejecutar el comando:

```
>regsvr32 <ruta del archivo>\MSWINSCK.OCX
```

Si se está en Windows 7, se debe ejecutar el CMD con privilegios de administrador.

Una vez copiada la carpeta, se deberán crear, si es que no existen, las carpetas “scripts” y “scripts”.

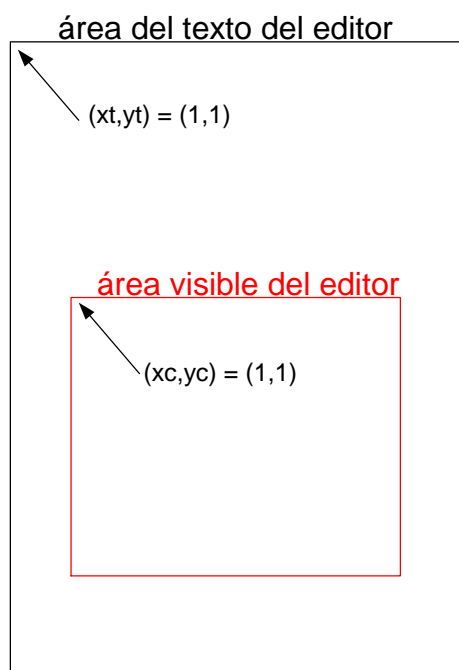
En la carpeta “scripts”, se guardarán los archivos de comandos que se usen en el programa. Estos archivos tienen extensión “.sh”.

En la carpeta “macros”, se guardarán los archivos de macros que se usen en el programa. Estos archivos tienen extensión “.ttm”.

COORDENADAS DEL EDITOR

A diferencia de la mayoría de terminales para Telnet, el Tito's Telnet Maneja de forma separada el Área Total del Texto, del Área del Terminal, del Área de la ventana visible.

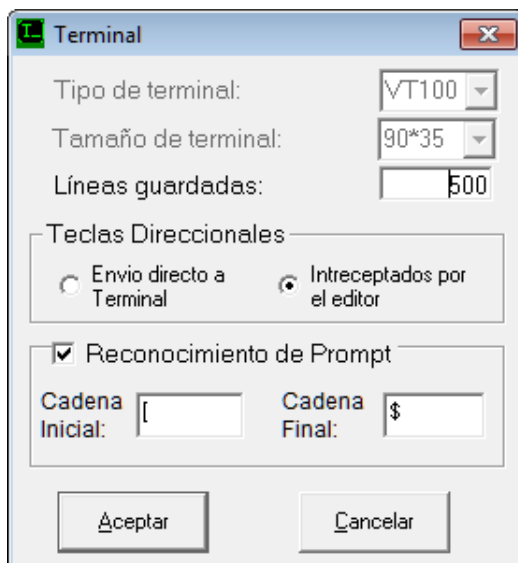
La ventana del terminal tiene es en realidad un visor de texto con funciones de terminal. El Área de texto del visor es por lo general mayor que el área de la ventana visible.



El Área visible depende del tamaño de la ventana principal de la aplicación. Se puede dimensionar en cualquier momento. No depende del tamaño del Área de texto total.

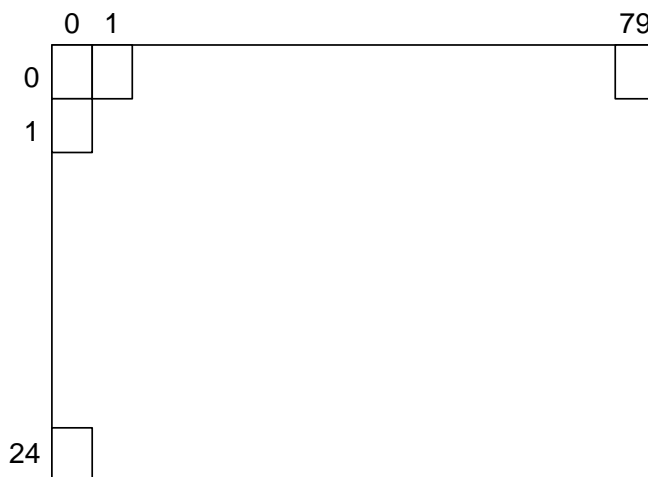
El área del texto total crece conforme van llegando caracteres por el terminal. Inicialmente es pequeña, pero va alcanzando varias líneas, hasta un máximo definido. A partir de allí, se van truncando las líneas superiores. El máximo de líneas que

soporta, se define en el campo “Líneas Guardadas”, en la ventana del menú “>Configuración>Terminal...”



COORDENADAS DE UN TERMINAL VT100 DE 80 * 25

Un terminal VT100 está diseñado a modo de pantalla única que está orientada a mostrar toda la información en un solo “pantallaza”. No está orientado a realizar desplazamientos.



Las coordenadas en un VT100 empiezan en la posición 0,0 y están siempre referidas a la ventana visible del terminal. Es decir que el punto 0,0 será siempre el

extremo superior izquierdo y el punto 79, 24 será siempre el extremo inferior derecho, aún después de los desplazamientos de pantalla.

Los límites de las coordenadas del terminal son fijos. No se cambian escribiendo texto ni redimensionando la ventana. Para cambiar el tamaño de una pantalla VT100 se debe negociar con el servidor un nuevo tamaño del terminal.

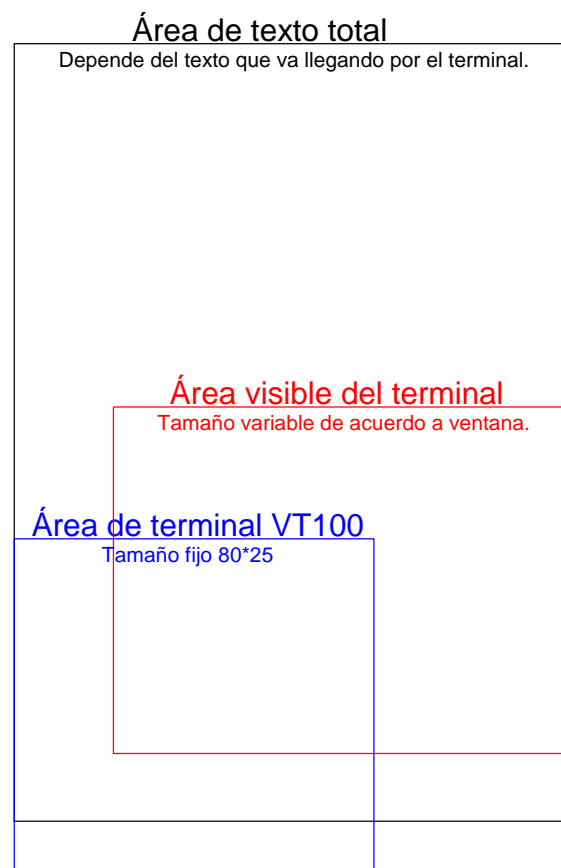
Este tamaño de pantalla es la única que manejan terminales simples como el “telnet” del DOS.

RELACIÓN ENTRE LAS 3 ÁREAS

En el Tito's Telnet existe un área especial para emular a un terminal VT100. Esta área es de tamaño fijo de 80 * 25 caracteres y se ubica casi siempre al final del área del texto total.

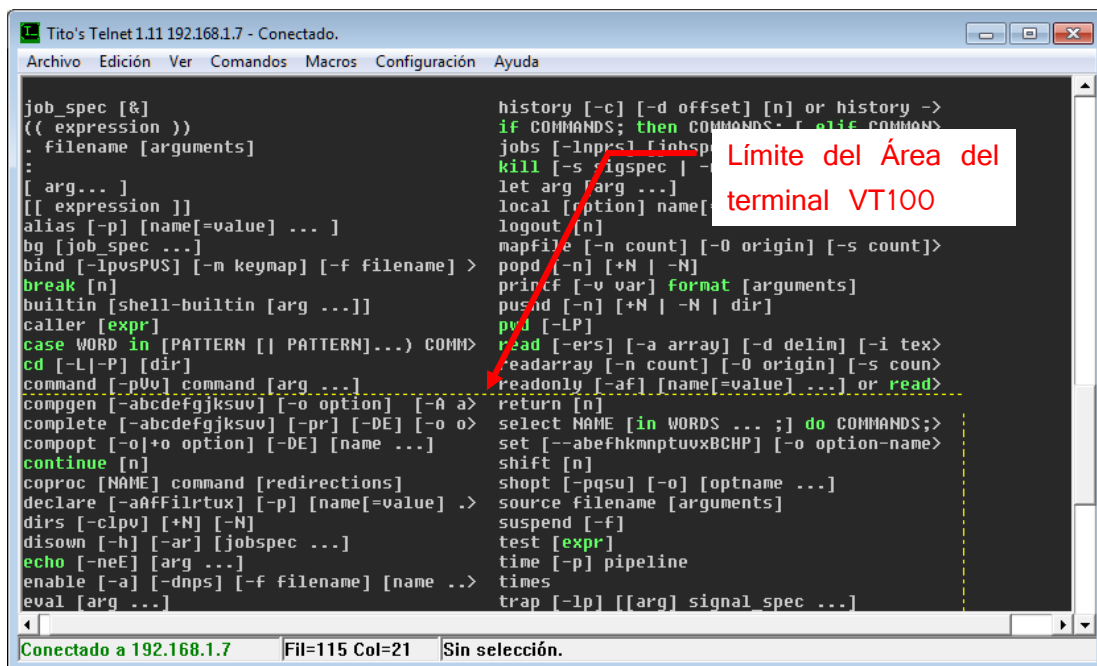
Para que el editor pueda manejar las coordenadas de un terminal VT100, se ha definido un nuevo sistema de coordenadas y además un nuevo cursor. Este nuevo sistema de coordenadas determina un área de trabajo que estará siempre pegada a la parte izquierda del área de las coordenadas de texto.

Una referencia a las 3 áreas se muestra en el gráfico:



La pantalla del VT100 siempre estará alineada al lado izquierdo del Área de texto total.

Tito's Telnet indica los límites de la ventana del terminal VT100 mediante unas marcas en la pantalla:



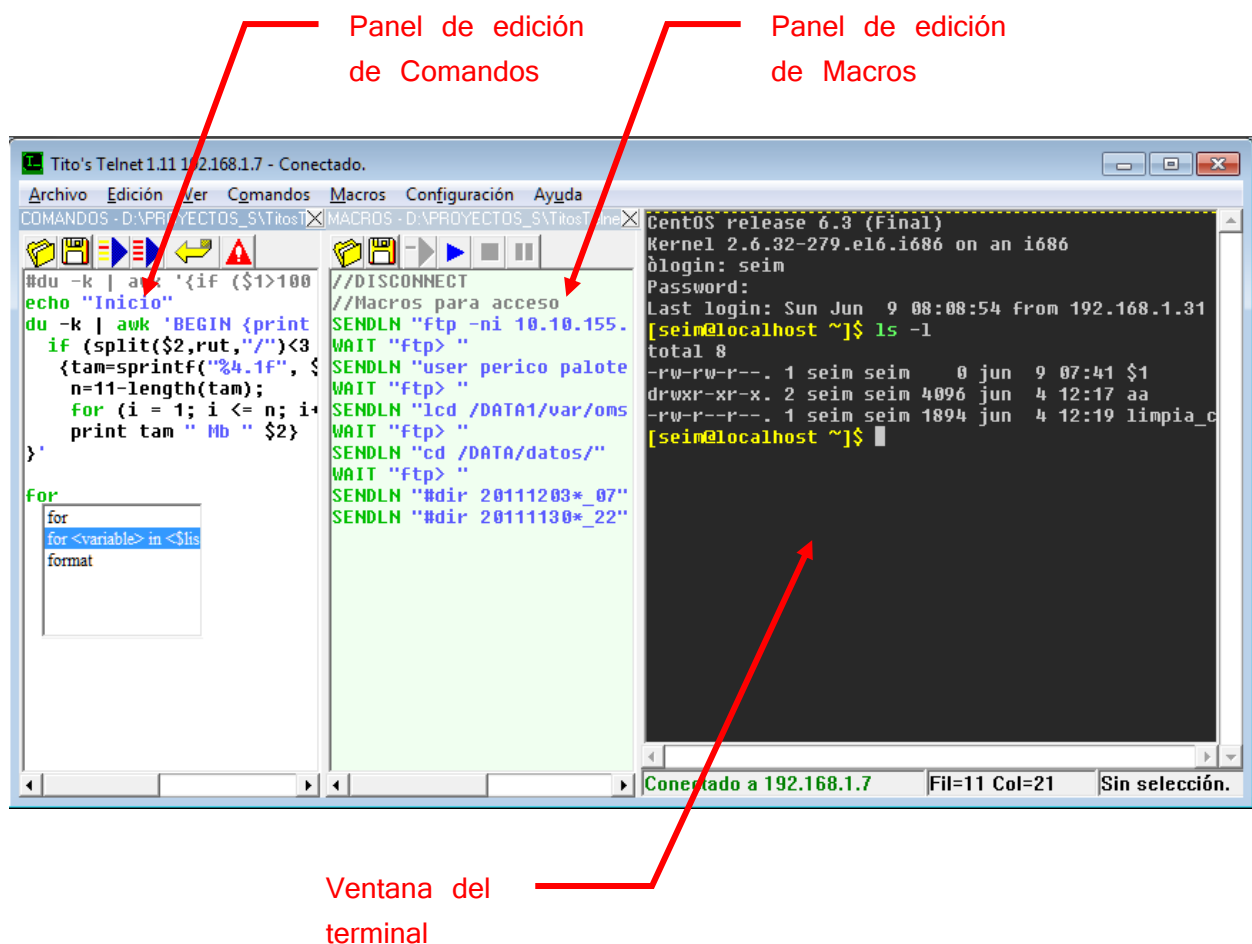
The screenshot shows the Tito's Telnet application window titled "Tito's Telnet 1.11 192.168.1.7 - Conectado.". The window has a menu bar with "Archivo", "Edición", "Ver", "Comandos", "Macros", "Configuración", and "Ayuda". The main area displays a list of shell commands in two columns. A red arrow points to a dashed yellow line that marks the bottom limit of the terminal window. A text box with the text "Límite del Área del terminal VT100" is placed over the arrow. The status bar at the bottom shows "Conectado a 192.168.1.7", "Fil=115 Col=21", and "Sin selección."

```
job_spec [&]
(( expression ))
. filename [arguments]
:
[ arg... ]
[[ expression ]]
alias [-p] [name[=value] ... ]
bg [job_spec ...]
bind [-lpusPUS] [-n keymap] [-f filename] >
break [n]
builtin [shell-builtin [arg ...]]
caller [expr]
case WORD in [PATTERN [| PATTERN]...] COMM>
cd [-L|-P] [dir]
command [-pUv] command [arg ...]
compgen [-abcdefgjkuv] [-o option] [-A a>
complete [-abcdefgjkuv] [-pr] [-DE] [-o o>
comptop [-o|+o option] [-DE] [name ...]
continue [n]
coproc [NAME] command [redirections]
declare [-aAfFiltux] [-p] [name[=value] .>
dirs [-clpv] [+N] [-N]
disown [-h] [-ar] [jobspec ...]
echo [-neE] [arg ...]
enable [-a] [-dnps] [-f filename] [name ..>
eval [arg ...]
history [-c] [-d offset] [n] or history ->
if COMMANDS; then COMMANDS; [ elif COMMANDS
jobs [-lnprsl] [jobspec ...]
kill [-s sigspec | -p pid | -t tpgid] [->
let arg [arg ...]
local [option] name[=value] ...
logout [n]
mapfile [-n count] [-O origin] [-s count]>
popd [-n] [+N | -N]
printf [-v var] format [arguments]
pushd [-n] [+N | -N | dir]
pwd [-LP]
read [-ers] [-a array] [-d delim] [-i tex>
readarray [-n count] [-O origin] [-s coun>
readonly [-af] [name[=value] ...] or read>
return [n]
select NAME [in WORDS ... ;] do COMMANDS;>
set [--abefhkmnptuvxBCHP] [-o option-name>
shift [n]
shopt [-pqsu] [-o] [optname ...]
source filename [arguments]
suspend [-f]
test [expr]
time [-p] pipeline
times
trap [-lp] [[arg] signal spec ...]
```

FUNCIONALIDADES

El cliente de telnet tiene varias herramientas adicionales al terminal. Estas son:

- Panel de Edición de Comandos
- Panel de Edición de Macros
- Editor Remoto.



Los paneles de comandos y de macros pueden activarse u ocultarse mediante el menú principal.

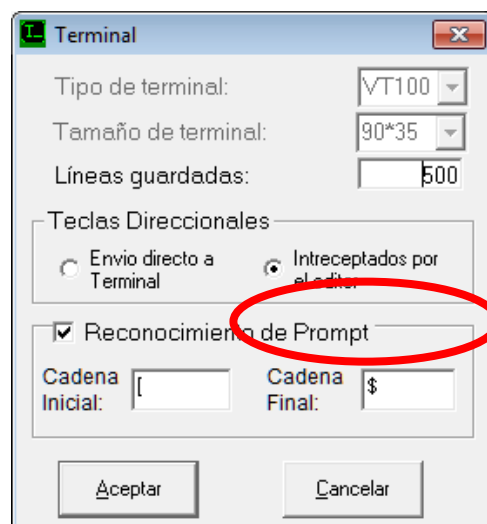
MANEJO DE CURSORES

Como se indicó anteriormente la aplicación maneja separadamente el Área total del texto y el Área correspondiente al terminal VT100.

De la misma forma, se maneja de forma separada, dos cursores. Uno corresponde al del terminal VT100, y lo controla la sesión telnet. Por lo general se encuentra al final de todo el texto. Sirve para editar los comandos que se introducen en la sesión y no se puede variar libremente su ubicación.

De manera separada se trata otro cursor que permite desplazarse libremente por todo el Área total del texto. Este cursor corresponde al visor del texto y su posición, determina la ubicación del Área visible del terminal.

Estos dos cursores suelen coincidir y mantenerse en la misma posición, pero se les puede “desenganchar” en cualquier momento, usando “mouse” para fijar la nueva posición del cursor libre. También se le puede separar, usando simplemente las teclas direccionales “arriba” o “abajo”, cuando está activada la opción “Teclas direccionales – Interceptados por el editor”, en la pantalla “Terminal”, del menú “>Configuración”



En este modo sin embargo, teclas direccionales “arriba” o “abajo” ya no serán recibidas por la sesión telnet.

Para hacer coincidir nuevamente los dos cursores, se deben posicionar en la misma coordenada, usando el “mouse” o usar la combinación de teclas <Ctrl>+<Fin>, que funcionará, siempre y cuando, el cursor del terminal VT100 esté al final del texto total.

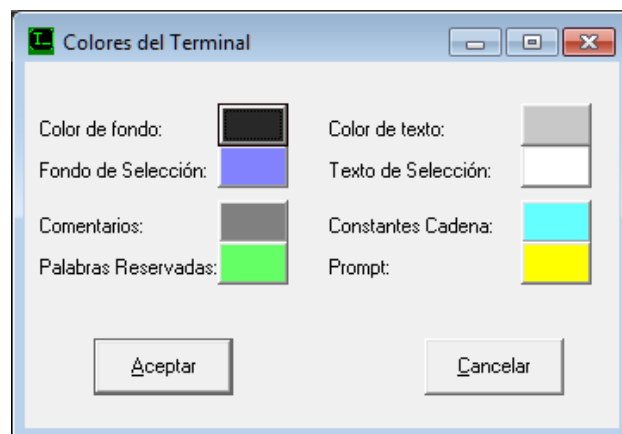
CARACTERÍSTICAS DEL TERMINAL

El terminal tiene la característica de coloreado de texto, en la pantalla del terminal. No se ha implementado el reconocimiento de las secuencias de color que suelen usar algunos “shell”. El coloreado lo hace el mismo programa, en base al reconocimiento de ciertas palabras y caracteres claves.

El lenguaje que se usa para el coloreado es el Shell de UNIX. Las palabras reservadas de este lenguaje, suelen ser los mismos que los comandos que se usan comúnmente en el trabajo diario con el terminal, como “echo”, “ls”.

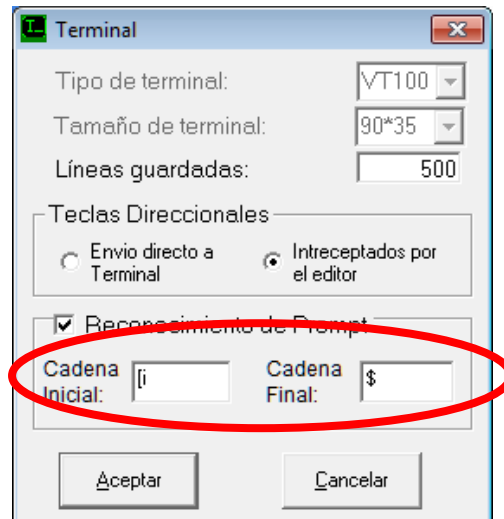
Se colorean los comentarios, palabras reservadas, constantes cadena y el “prompt”.

Para cambiar las configuraciones colores usar la ventana “colores”, desde el menú “>Configuración”:



El coloreado del prompt, no se hace de forma directa, sino que se debe realizar primero el reconocimiento del “prompt” en el terminal. Para ello se usa un método que consiste en detectar los caracteres iniciales y finales. Este método no es seguro, pero funcionará bien en la mayoría de configuraciones del “prompt”.

Si no se puede detectar correctamente el “prompt”, se deberá elegir otro “prompt” o elegir otro grupo de caracteres en la ventana de configuración:



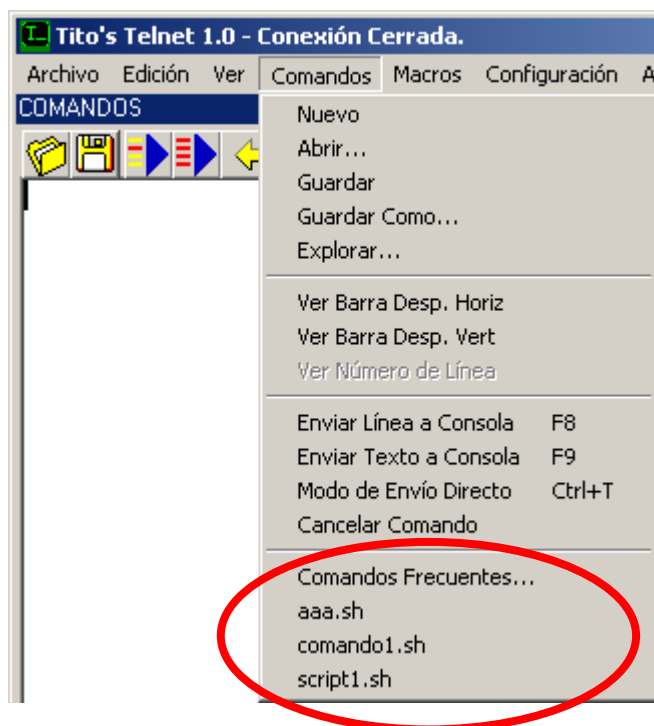
Tener bien configurado el reconocimiento del “prompt”, es importante no solo por fines de coloreado, sino que también ayuda para el trabajo del Editor de comandos (Ver más adelante) y el trabajo de las macros.

EDITOR DE COMANDOS

El panel de edición de comandos es el panel lateral izquierdo, que permite escribir los comandos del “Shell”, en un editor antes de enviarlos al terminal.

Este texto se puede guardar en disco en un archivo de comandos o “script”, para usarlos posteriormente. Por defecto los archivos de comandos se guardan en la carpeta “/scripts”, en la carpeta principal del programa.

Los archivos de comandos también se pueden ejecutar directamente desde el menú “Comandos”, seleccionando el nombre del archivo. Los archivos de comandos en la carpeta “/scripts” se cargan en este menú de forma automática al iniciar el programa.



El editor de comandos es un editor normal de texto con algunas características adicionales como el resaltado de sintaxis y ayuda contextual. El editor permite además enviar el texto al terminal de tres formas:

1. Enviar todo el texto del editor.
2. Enviar solo el texto seleccionado.
3. Enviar la línea actual. Es el modo de “envío directo”.

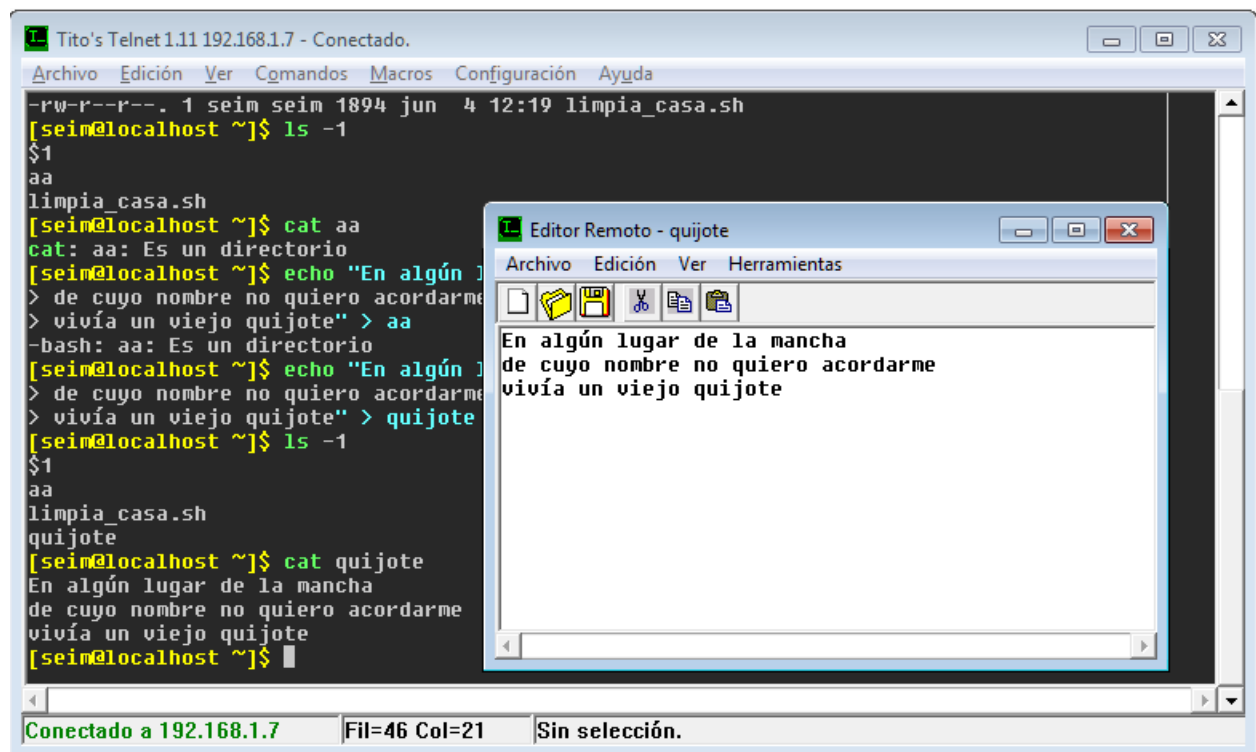
El panel de Edición de macros permite trabajar con las macros de manera similar a como se trabajan con los archivos de comandos.

Por defecto los archivos de macros se guardan en la carpeta “/macros”, en la carpeta principal del programa.

También se pueden ejecutar directamente las macros desde el menú “Macros”, seleccionando el nombre de la macro.

EDITOR REMOTO

El editor remoto es una funcionalidad que permite editar en una ventana de edición, un archivo de texto en el servidor. La edición se hace en el “editor remoto” del programa y se comunica con el servidor mediante comandos “telnet” de forma transparente para el usuario.



Para que estén activas las opciones de edición remota es necesario que esté activada y configurada la opción “Reconocimiento de prompt”, en el menú “Configuración>Terminal...”.

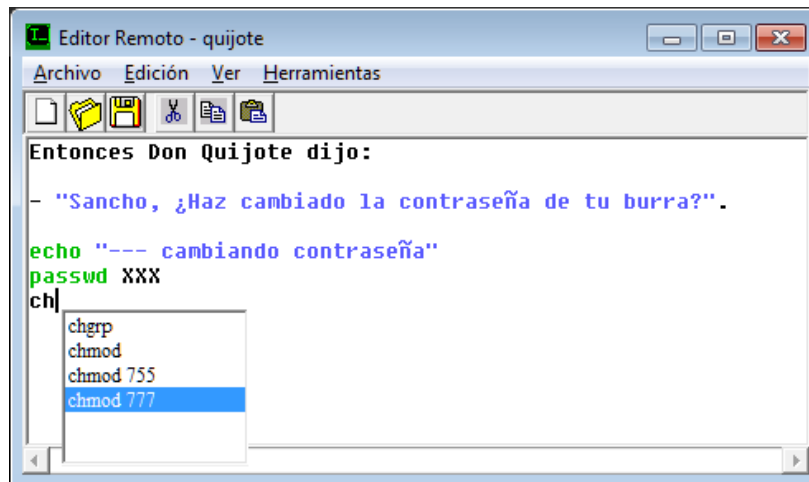
Si el sistema no puede reconocer el “prompt”, no se podrán usar las opciones de edición remota. Además de acuerdo al tipo de shell y sistema operativo, podría no lograrse la edición remota de archivos.

Como las capacidades para mover información al servidor remoto mediante telnet, son limitadas, no es factible para la edición de archivos grandes. Solo se recomienda el uso para archivo de 100 líneas o menos, pero dependerá también de la velocidad de la red.

El editor remoto se puede abrir de forma abreviada usándola combinación de teclas <Ctrl>+<F2>.

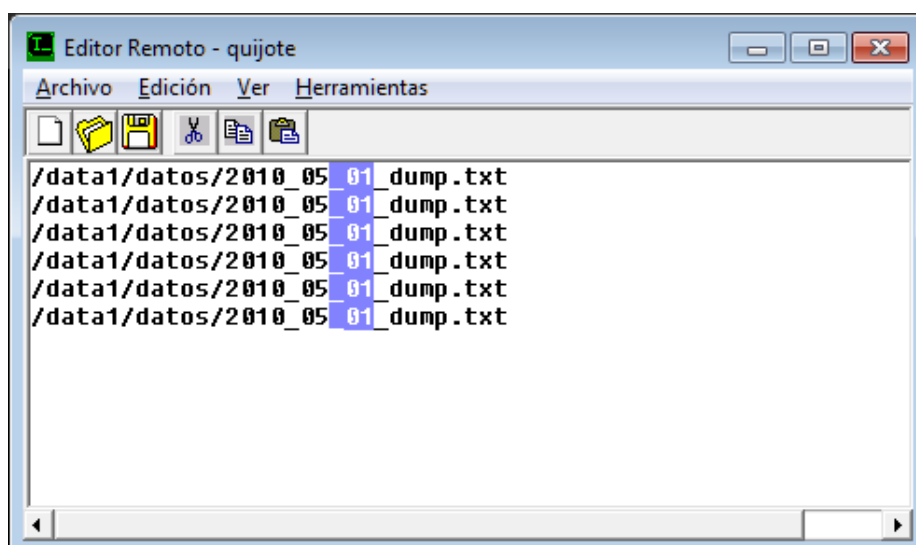
Una vez abierto el editor remoto, es posible usar las opciones “Abrir”, “Guardar” y “Guardar como...”, del menú del editor remoto.

El editor remoto tiene incluido la opción fija de coloreado de sintaxis, para comandos del Shell. No se puede cambiar este lenguaje de trabajo.



También existe una ayuda de contexto, texto inteligente que ayuda en la creación de programas del shell.

Es posible también trabajar en modo de columnas tanto para selección como para edición.



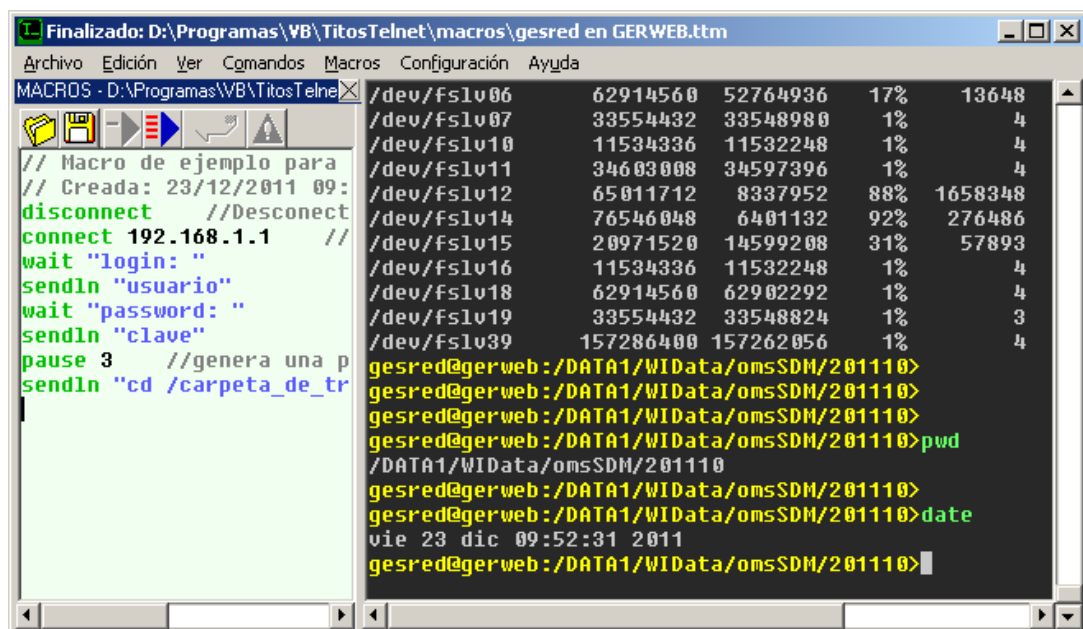
Al modo columna se entra a través del menú “Edición” o mediante el método abreviado “Alt-C”.

AUTOMATIZACIÓN

El terminal soporta la ejecución de macros programados en un sencillo lenguaje de programación de alto nivel.

El lenguaje de programación está basado en el lenguaje del terminal TeraTerm con algunas diferencias.

Las macros se pueden crear en un editor normal de texto o se pueden editar en el panel de macros que incluye un editor con resaltado de sintaxis y ayuda contextual.



The screenshot shows the Tito's Telnet application interface. The top window is titled "Finalizado: D:\Programas\VB\TitosTelnet\macros\gesred en GERWEB.ttm". Below it is a menu bar with "Archivo", "Edición", "Ver", "Comandos", "Macros", "Configuración", and "Ayuda". The main area is split into two panes. The left pane, titled "MACROS - D:\Programas\VB\TitosTelnet", contains a macro script with comments and commands:
`// Macro de ejemplo para`
`// Creada: 23/12/2011 09:`
`disconnect //Desconect`
`connect 192.168.1.1 //`
`wait "login: "`
`sendln "usuario"`
`wait "password: "`
`sendln "clave"`
`pause 3 //genera una p`
`sendln "cd /carpeta_de_tr`
The right pane shows a terminal window with the following output:
`/dev/fs1v06 62914560 52764936 17% 13648`
`/dev/fs1v07 33554432 33548980 1%`
`/dev/fs1v10 11534336 11532248 1%`
`/dev/fs1v11 34603008 34597396 1%`
`/dev/fs1v12 65011712 8337952 88% 1658348`
`/dev/fs1v14 76546048 6401132 92% 276486`
`/dev/fs1v15 20971520 14599208 31% 57893`
`/dev/fs1v16 11534336 11532248 1%`
`/dev/fs1v18 62914560 62902292 1%`
`/dev/fs1v19 33554432 33548824 1%`
`/dev/fs1v39 157286400 157262056 1%`
`gesred@gerweb:/DATA1/WIData/omsSDM/201110>`
`gesred@gerweb:/DATA1/WIData/omsSDM/201110>`
`gesred@gerweb:/DATA1/WIData/omsSDM/201110>`
`gesred@gerweb:/DATA1/WIData/omsSDM/201110>pwd`
`/DATA1/WIData/omsSDM/201110`
`gesred@gerweb:/DATA1/WIData/omsSDM/201110>`
`gesred@gerweb:/DATA1/WIData/omsSDM/201110>date`
`vie 23 dic 09:52:31 2011`
`gesred@gerweb:/DATA1/WIData/omsSDM/201110>`

Desde el panel de macros se pueden crear, modificar y ejecutar las macros. Las macros se guardan por defecto en la carpeta “/macros” en la carpeta principal del programa.

Las macros también se pueden ejecutar directamente desde el menú “Macros”, seleccionando el nombre de la macro. Las macros en la carpeta “/macros” se cargan en este menú de forma automática al iniciar el programa.

VARIABLES Y TIPO DE DATOS

Las variables son elementos del lenguaje que permiten almacenar valores de tipo numérico o de cadena.

No es necesario declarar las variables antes de usarlas dentro del programa.

Los nombres de las variables pueden tener solo caracteres alfanuméricos pero no pueden empezar con un dígito numérico.

El lenguaje solo diferencia dos tipos de datos:

1. Números
2. Cadenas

El sistema hará conversiones automáticas de tipo de acuerdo al contexto de las operaciones que se realice.

Las fechas y los valores boléanos se tratan como número.

Las asignaciones se hacen con el símbolo: =

X=1;

ESTRUCTURA DEL LENGUAJE

Un programa está dividido en instrucciones. Las instrucciones se pueden separar por saltos de línea o por el punto y coma.

Aunque no es obligatorio, es recomendable el uso del punto y coma como separador de instrucciones para evita ambigüedades.

Los comentarios solo pueden ser de una línea y empiezan con los caracteres “//”.

Todas las instrucciones se consideran también como expresiones.

REFERENCIA DE INSTRUCCIONES

CONNECT

Abre una sesión Telnet a una dirección específica.

DISCONNECT

Termina la sesión Telnet que está en curso.

SENDLN

Envía una cadena al terminal agregando un salto de línea al final.

WAIT

Espera a que aparezca una cadena específica en el terminal.

PAUSE

Genera un retardo de tiempo antes de continuar con la ejecución del programa.
El tiempo se da en segundos.

STOP

Termina la ejecución de la macro actual.

END

Termina el aplicativo. Cierra el programa.

LOGOPEN

Abre un archivo de texto para usarlo como registro de la salida del terminal.

LOGWRITE

Escribe una cadena en el archivo de registro.

LOGCLOSE

Cierra el archivo de registro actual.

LOGPAUSE

Pausa la escritura de datos en el archivo de registro.

LOGSTART

Reinicia la escritura de datos en el archivo de registro.

FILEOPEN

Abre un archivo para leer o escribir datos.

FILECLOSE

Cierra un archivo que estaba abierto previamente.

FILEWRITE

Escribe datos en un archivo abierto previamente con FILEOPEN.

MESSAGEBOX

Muestra un mensaje en pantalla en una pequeña ventana modal. La ejecución del programa se detiene hasta que el usuario pulse "Aceptar" en el mensaje de texto.

CAPTURE

Inicia la captura de la salida del terminal en una variable cualquiera.

ENDCAPTURE

Finaliza la captura iniciada con CAPTURE

EDIT

Lanza el editor remoto para editar algún archivo del servidor.

ESTRUCTURA CONDICIONAL

Las estructuras condiciones son de l tipo:

IF ... ENDIF

IF ... ELSE ... ENDIF

Las expresiones a usar se consideran como verdaderas cuando devuelven un valor diferente de cero y se consideran falsas cuando devuelven cero.

EXPRESIONES

Las expresiones están compuestas de operandos y operadores.

Los operadores reconocidos por orden de jerarquía son:

JERARQUÍA	OPERADOR
1	">>", "<<", ">+", ">-"

2	"="
3	"&&", " ", "!", "!!"
4	"==", "<>", ">", ">=", "<", "<=", "~"
5	"+", "-", " ", "&"
6	"*", "/", "\", "%"
7	"=<", "=>"
8	"^", "++", "--", "+=", "-=", "*=", "/="

Tabla de contenido

NOTAS SOBRE EL TITO'S TELNET.....	2
¿POR QUÉ UN CLIENTE MÁS?	3
INSTALACIÓN.....	5
COORDENADAS DEL EDITOR.....	6
COORDENADAS DE UN TERMINAL VT100 DE 80 * 25	7
RELACIÓN ENTRE LAS 3 ÁREAS	8
FUNCIONALIDADES	10
MANEJO DE CURSORES	10
CARACTERÍSTICAS DEL TERMINAL	12
EDITOR DE COMANDOS	14
EDITOR REMOTO	16
AUTOMATIZACIÓN.....	19
VARIABLES Y TIPO DE DATOS.....	20
ESTRUCTURA DEL LENGUAJE.....	20
REFERENCIA DE INSTRUCCIONES	21
Tabla de contenido.....	24