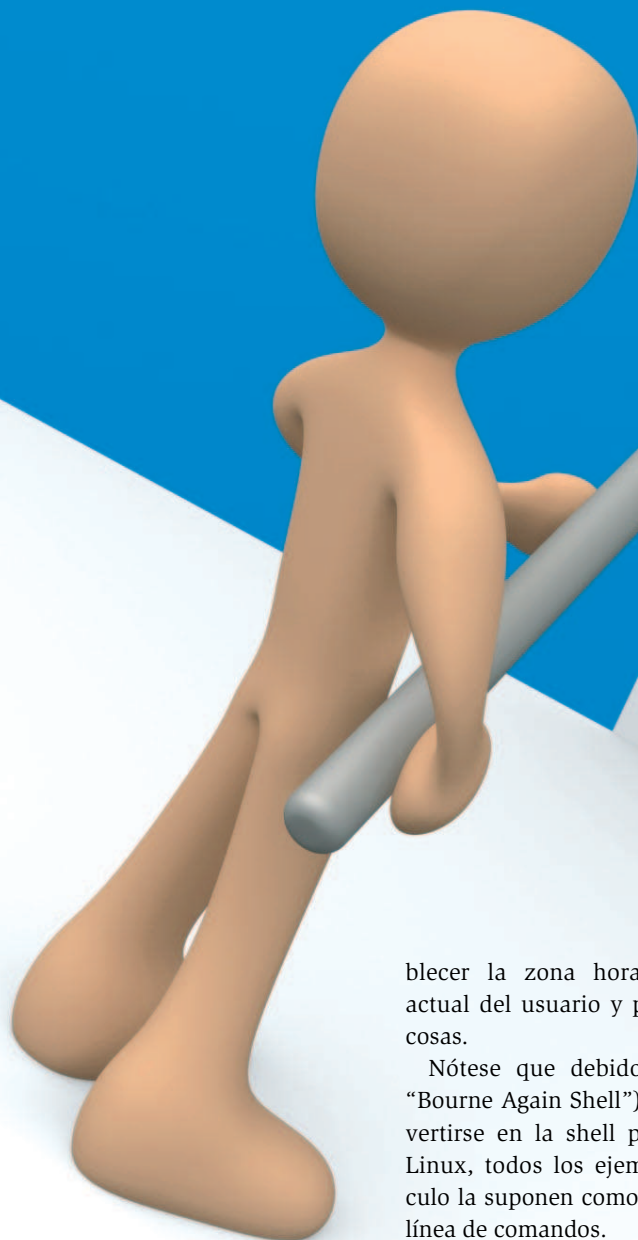


Configuración de la shell con variables de entorno

# HECHO EN CASA



Las variables de entorno nos permiten configurar la línea de comandos para adaptarla a nuestras propias necesidades. **POR HEIKE JURZIK**

blecer la zona horaria, el directorio actual del usuario y para muchas otras cosas.

Nótese que debido a que bash (la “Bourne Again Shell”) ha llegado a convertirse en la shell predeterminada de Linux, todos los ejemplos de este artículo la suponen como el intérprete de la línea de comandos.

Una variable de entorno tiene tres partes: el nombre, un operador de asignación (=) y un valor:

```
variable=valor
```

El nombre puede incluir letras mayúsculas y minúsculas, subrayados y números. La única restricción es que una tabla no puede comenzar con un número. Si el valor de la variable contiene espacios en blanco o caracteres no estándares, deberemos entrecomillar nuestra entrada:

```
LESS=" -X"
```

Dentro de un sistema Linux cada usuario tiene un lugar al que puede llamar su hogar: su directorio *home*. Bash proporciona un entorno completamente amueblado para cada cuenta única, aunque la apariencia dependerá de la distribución que se use. Para conocer las variables que usa nuestro home escribimos *env* (de “environment” o entorno). El Listado 1 muestra un ejemplo.

Algunas de las variables de este Listado pueden ser configuradas por el usuario. Otras se encuentran predefinidas, aunque podemos modificarlas cuando sea necesario. La convención exige usar letras mayúsculas. En la Tabla 1 encontramos algunas de las variables más comunes para bash.

El comando *echo* nos permite descubrir el valor de la variable, presentándolo

**S**i introducimos un comando como *ls* en la shell, ésta sabe que estamos llamando al programa */bin/ls*. Bash conoce dónde buscar los ejecutables en el sistema de archivos, de manera que no es necesario escribir la ruta completa. Esta ruta se encuentra definida en lo que se conoce como una variable de entorno. Variables de este tipo también se usan para modificar la apariencia del prompt de la shell, esta-

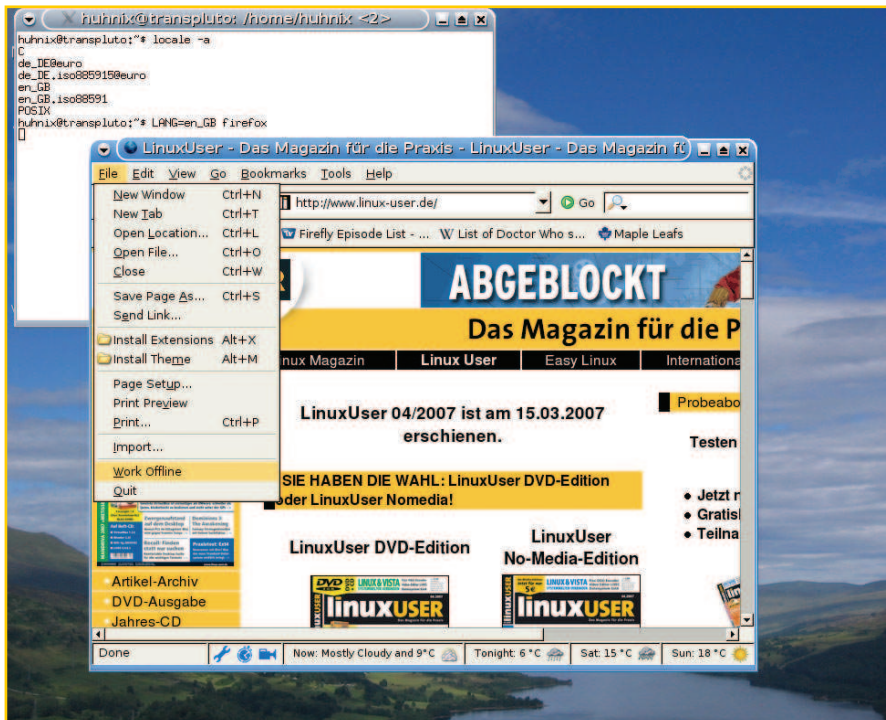


Figura 1: Podemos cambiar las variables temporalmente sin preocuparnos de los efectos posteriores

de manera correcta en la línea de comandos:

```
huhn@asteroid:~$
echo Huhu
Huhu
```

Además, *echo* presenta el valor de la variable si escribimos el signo dólar delante del nombre de la variable:

```
huhn@asteroid:~$
echo $LANG
es_ES@euro
huhn@asteroid:~$
echo $PS1
\u@\h:\w\$
```

### TRUCO

En vez de usar dos comandos separados (por ejemplo, *LESS="-X"; export LESS*) podemos definirlos y exportarlos de un tirón: *export LESS="-X"*.

### GLOSARIO

**Builtin:** Abreviatura del comando "built-in" (o incorporado). Existen comandos de este tipo incorporados en la shell, y no necesitamos ejecutar ningún programa especial para usarlos.

### Listado 1: Ejemplo env

```
01 huhn@asteroid:~$ env
02 TERM=xterm
03 SHELL=/bin/bash
04 USER=huhn
05 LS_COLORS=no=00:fi=00:...
06 PATH=/usr/local/bin:/usr/
07 bin:/bin:/usr/bin/X11:/usr/
08 games:/home/huhn/bin
09 IRCSERVER=irc.freenode.net
10 LANG=es_Es@euro
11 ...
```

grama, el ejemplo enseña cómo obtener una lista de los locales que soporta nuestra máquina con el comando *locale -a*.

### Cambiar

Como hemos podido ver, es fácil cambiar una variable introduciendo su nombre, el operador de asignación y el valor. Las variables definidas de este modo solamente son válidas para la sesión de terminal actual. Para exportar la configuración necesitamos el comando *export*. Esto hace que las configuraciones sean accesibles a los procesos hijos y subshells.

Por ejemplo, podemos configurar las variables *IRC\_NICK*, *IRC\_NAME* y *IRC\_SERVER* para que el bash actual le diga a los clientes IRC basados en la línea de comandos, tales como Irssi o BitchX, el nick y el nombre que deben usar para que registren automáticamente un servidor IRC específico. Si luego arrancamos una segunda shell de la primera e intentamos presentar los valores de las nuevas variables, no veremos nada. Con *exit* (Ctrl + D) abandonamos la segunda shell y

La shell reemplaza las variables *\$LANG* y *\$PS1* por sus respectivos valores y luego llama a *echo* para que presente el texto.

Incluso si tenemos todo configurado tal y como nos gusta, siempre está bien cambiar de la rutina diaria. Por ejemplo, si quisiéramos refrescar nuestras herramientas de lenguaje, podríamos usar la siguiente línea de comandos para arrancar el navegador Firefox en Francés, aún cuando nuestro entorno normalmente hable Inglés:

```
LANG=FR firefox
```

La Figura 1 muestra cómo cambiar temporalmente a un entorno de lenguaje diferente. Antes de llamar al pro-

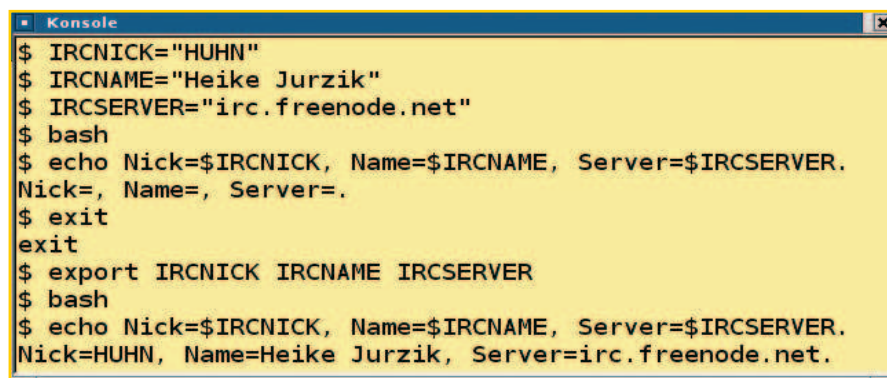


Figura 2: El comando export se usa para exportar nuevas variables a entornos subshell.

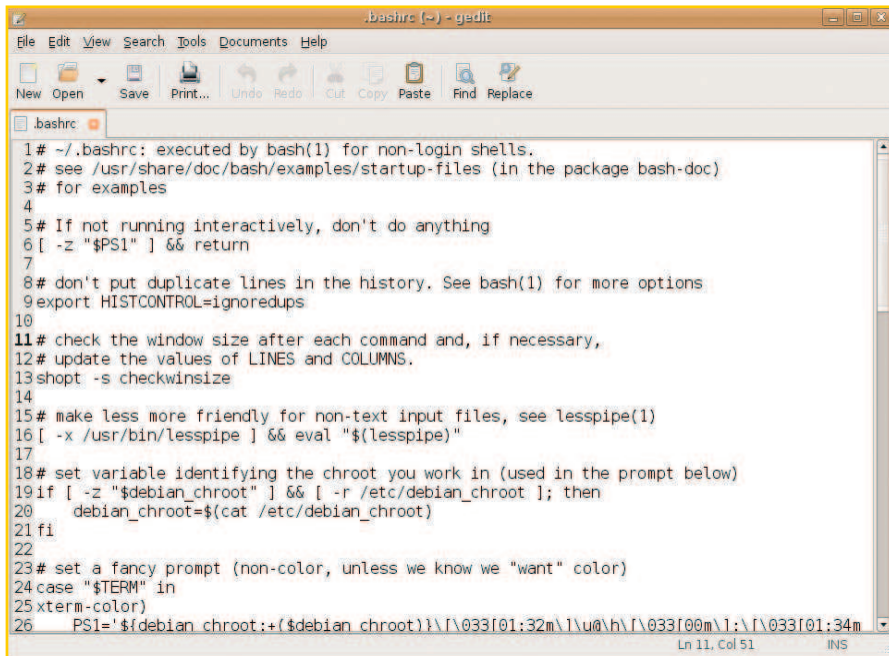


Figura 3: El fichero bashrc contiene las opciones de configuración para el entorno bash.

exportamos las nuevas variables. A continuación abrimos una segunda shell a partir de la primera, y podremos comprobar que ha heredado variables que ahora se encuentran disponibles en la subshell (Figura 2).

El *export* incorporado guarda más cartas en su manga. En combinación con la opción *-p*, o cuando es invocado

sin ningún parámetro, presenta una lista de todas las variables exportadas. Para configurar una variable permanentemente necesitamos modificar el fichero de configuración bash y volver a cargar su entorno. Para ello añadimos el nuevo comando *export* a uno de los ficheros de arranque de bash, como *~/.bashrc* (Figura 3), y luego lo

analizamos una vez hayamos acabado nuestras mejoras:

```
source ~/.bashrc
```

El listado 3 muestra un par de variables de entorno prácticas, que deberían hacernos sentir como en casa. ■

### Listado 2: Ejemplo de export

```
01 huhn@asteroid:~$ export
02 declare -x CHARSET="latin1"
03 declare -x LS_COLORS="no=00:
04 fi=00:..."
05 declare -x IRC_SERVER="irc.
06 freenode.net"
07 ...
08 huhn@asteroid:~$ export -n
09 IRC_SERVER
```

### Listado 3: Variables de Entorno

```
01 # evita que "less" deje una
02 # pantalla vacía al final de
03 # la salida:
04 export LESS=""-X"
05 # añade un toque de color:
06 # prompt del
07 # usuario en verde, el de root
08 # en rojo:
09 if [ $(id -u) = 0 ] ; then
10 COLOR1='\[\033[00;31m\'
11 else
12 COLOR1='\[\033[01;32m\'
13 fi
14 COLOR2='\[\033[00;33m\'
15 COLOR3='\[\033[00;37m\'
16 PS1=$COLOR1'\[u@\h'$COLOR2'
17 \w'$COLOR1']'$COLOR3'\$ '
18 PS2=$COLOR1'>'$COLOR3' '
19 # quiero la zona horaria CET
20 # para mi servidor
21 # en Canadá
22 export TZ=CET
23 # Quiero que ispell utilice el
24 # juego de
25 # caracteres y el diccionario
26 # correctos:
27 export DICTIONARY=nspanish
28 export CHARSET=latin1
```

Tabla 1: Variables de Entorno Bash Estándares

| Variable    | Significado   |
|-------------|---|
| CDPATH      | Busca la ruta para el comando cd  |
| EDITOR      | Editor de texto estándar  |
| HISTFILE    | Fichero histórico (e.g., ~/.bash_history)   |
| HISTSIZE    | Máximo número de comandos a guardar en el histórico   |
| HOME        | Directorio home para la cuenta de usuario   |
| HOSTNAME    | Nombre de Host  |
| LANG        | Lengua para salidas del programa, tales como el formato de fecha, etc., suponiendo que ninguna de las variables LC_ (ver abajo) han sido establecidas. Podemos obtener una lista de todas las variables de lengua definidas mediante el comando <i>locale</i> |
| LC_ALL      | Establece el país, como <i>C</i> o <i>de</i> ; esto invalida LANG y cualquier otra variable LC_ variable  |
| LC_MESSAGES | Lengua del programa y de los mensajes de error  |
| LC_TIME     | Formato de la hora/fecha  |
| LOGNAME     | Login del nombre de usuario   |
| MAIL        | Ruta al correo de usuarios (entrada de correo)  |
| MANPATH     | Busca las rutas para las páginas man  |
| PATH        | Busca la ruta para los ejecutables  |
| PS1         | Apariencia predeterminada para el prompt de la shell. Para usuarios sin privilegios, usualmente es <i>\u@\h:\w\</i> \$ (huhn@asteroid:~\$); para el administrador, usualmente es <i>\h:\w\</i> \$ (p.e.: asteroid:~#).  |
| PWD         | Nombre del directorio actual ("print working directory")  |
| SHELL       | Nombre de la ruta completa de la shell actual (/bin/bash)   |
| TERM        | Opciones del tipo de terminal, tales como <i>xterm</i> o <i>vt100</i>   |
| TZ          | Zona horaria, como CET o MET  |