

Elogio de la diversidad lingüística

UN TECLADO DE 60.000 TECLAS

En el número anterior aprendíamos a *localizar* nuestra interfaz de usuario, y a instalar los tipos de letra que necesitamos. Hemos instalado un tipo unicode y elegido un *locale* UTF-8, recordemos que argumentábamos que el uso predominante de iso-8859-1 es un signo de dominación de Occidente.

POR JUAN RAFAEL FERNÁNDEZ

Incluso la distinción entre mayúsculas y minúsculas, que nos parece tan natural, es un fenómeno infrecuente entre las escrituras del mundo: leo en Nick Nicholas, http://ptolemy.tlg.uci.edu/~opoudjis/unicode/unicode_gkbkgd.html, que habiendo el georgiano moderno abandonado hace ya mucho su uso medieval de la distinción, hoy día está limitada al griego, latín, armenio y cirílico. Prometíamos entrar de lleno en la cuestión más ardua: cómo conseguir con las tres filas de teclas de nuestro teclado introducir los miles de caracteres necesarios para escribir los distintos lenguajes. Ha llegado el momento de responder a esta pregunta.

Creo haber escrito ya suficientes veces que la historia del software libre explica que las soluciones a casi todos los problemas sean diversas. No pretenderé en este artículo ser exhaustivo, ni aburrido, recordemos la vocación práctica de la que nace esta sección.

Como a los problemas casi se les puede poner cara y circunstancia (el profesor de griego clásico, el tutor de alumnas chinas, el traductor del ruso) las respuestas irán de lo particular a lo general. Puesto que ningún teclado contiene los miles de caracteres necesarios para escribir los miles de idiomas del mundo, es necesario alguna forma de «mapear» los diferentes alfabetos en nuestro teclado. Como decía el poema, diversas son las soluciones como diversos los hombres. O algo así.

Thessalonika para OpenOffice, o el espíritu clónico

Mi amigo el profesor de griego sólo quería unas macros para poder escribir griego politónico (con acentos, espíritus, iota suscrita...) en su OpenOffice, porque echaba de menos las que utilizaba en Word. Como el alfabeto griego contemporáneo es monotónico (hay

guerras de religión sobre el tema pero aquí se nos permitirá la simplificación) no basta elegir un teclado griego, hay que poder introducir el resto de los signos y combinarlos. ¿Se puede? Pues sí, la solución se llama **Thessalonika**, la podemos descargar libremente de <http://www.thessalonika.org.ru/en> y la ha creado Alexej Kryukov.

Antes de entusiasrnos debemos decir que *Thessalonika* está escrito en java, lo cual no es pecado en sí mismo pero hace que no funcione en las distribuciones como Debian y derivadas que únicamente utilizan software radicalmente libre. Si necesitamos y queremos usar Thessalonika deberemos descargar una versión de OpenOffice de <http://www.openoffice.org> e instalarla por nuestra cuenta; también nos hará falta el ejecutor de aplicaciones java *jre*. La documentación de Thessalonika está traducida al castellano y la instalación es trivial, poner el archivo en el subdirectorio adecuado y regenerar la cache de añadidos.

¿Qué nos aporta? La posibilidad (ver figura 1) de alternar hasta tres tipos de teclados de escrituras diferentes (por defecto latina, griega politónica y cirílica). Como estamos ayudando a un profesor de griego pondremos en el cuadro 1 una chuleta sobre la forma de introducir los caracteres griegos, y escribiremos un ejemplo probatorio, la figura 2.

Interesante, pero limitado: ni sólo se escribe en OpenOffice (aunque cueste creerlo) ni los únicos alfabetos no latinos son el griego y el cirílico. Y nos obliga a abandonar los paquetes estándar y Stallman nos puede retirar el saludo... necesitamos una solución más general.

Yo me lo guiso, yo me lo como: yudit

La siguiente solución que analizaremos la proporciona el editor de textos unicode (no procesador de textos) **yudit**, del húngaro Gáspár Sinai. *yudit* es el programa clásico que tuvo que solucionar por su cuenta todos los problemas a los que se enfrentaba: cómo escribir unicode, cómo imprimir textos en unicode. Y los solucionó bien. La configuración permite integrar multitud de tipos de letra específicos que deberemos ir instalando. Como método de entrada, que debía bastar para el limitado teclado americano, ideó métodos de mapeo de teclado para casi cualquier alfabeto. Podemos consultar en `/usr/share/yudit/src/` los mapas; en nuestro caso estudiaremos *GreekPolytonic.kmap*. Unas cuantas líneas bastarán

```
":a=0x03AC", // GREEK SMALL
LETTER ALPHA WITH TONOS
```

que significa que un punto y como seguido de una «a» produce el carácter «ᾶ». Como el recortar y pegar funciona perfectamente entre yudit y gedit no es necesario mostrar capturas sino que podemos integrar la salida en nuestro texto. Sigamos

">a=0x1F00", // GREEK SMALL
LETTER ALPHA WITH PSILI = «ᾶ»
"<a=0x1F01", // GREEK SMALL
LETTER ALPHA WITH DASIA = «ᾷ»

¿Cómo se escribe en árabe? Pues seleccionando un mapa de teclado árabe, como *Arabic.kmap*. Sirva las figuras 3 como ejemplo y prueba. Quiero terminar el tratamiento de yudit mostrando la posibilidad de escribir a mano alzada caracteres CJK (figura 4). Y la salida impresa de los documentos escritos con yudit es perfecta, lo que no siempre ocurre.

Pero, por muy bueno que sea yudit, no puede bastarnos: necesitamos poder escribir caracteres unicode en todos los programas, en los formularios del navegador, en las bases de datos... ¿Qué hacemos?

Los planos de un teclado

La relación humeana entre la pulsación de una tecla que tiene pintada la letra «a» y que en la pantalla del monitor se dibuje un conjunto de pixeles que reconozco como la misma letra dista de ser simple; cuando pulsamos una tecla estamos produciendo una señal eléctrica *en crudo* que es convertida por el sistema operativo a un número, el código de tecla o *scan-code*, que de forma configurable es traducida a un símbolo (el *keysym*). Este camino que hemos iniciado nos lleva demasiado lejos, y sólo vamos a señalarlo para el lector curioso; la figura 5 cómo puede configurarse. Quien quiera saber más puede leer con provecho el manual de *xkeycaps*.

A donde queríamos llegar con cuidado de no perdernos es a señalar que hace años que es posible configurar con todo detalle el comportamiento del teclado. Se ha reconocido que era un problema que debía ser resuelto de forma estándar y coherente y ya estamos en condiciones de comprender por qué la configuración puede realizarse en distintas capas: varios agentes han trabajado en la solución, en distintos planos superpuestos; a muy bajo nivel, con la extensión *xbk* al protocolo X; en las capas de los entornos de escritorio, con la estandarización promovida por Gnome o KDE; o al nivel de las aplicaciones como habíamos visto.

Retomemos el hilo, queríamos ver cómo se configura el teclado y si podíamos alternar mapas de teclado para poder escribir consecutivamente en

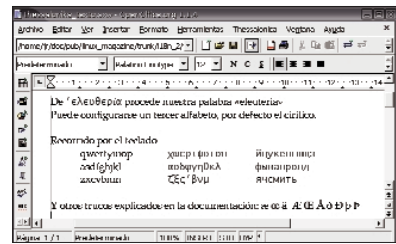


Figura 2: Thessalonika en pleno uso.

castellano y griego. No queremos una solución esporádica (sé por ensayo y error que la combinación de las teclas *Alt-Gr* y *z* produce el signo tipográfico «*«*» —también podía haberlo mirado en `/etc/X11/xkb/symbols/es—`, pero sé también que las combinaciones de teclas que puedo hacer y puedo recordar son limitadas) ni para una sola aplicación (sería absurdo escribir con yudit y estar pegando al resto de las aplicaciones), sino una solución que valga para todos los programas. ¿Existe? Sí. Y la tenemos delante de nuestras narices desde el principio.

Cualquiera que haya buceado en GNU Linux deberá haberse familiarizado con

Tabla 1: Asignaciones de teclas

Tecla	Letra conseguida
[a]	α
[n]	ν
[b]	β
[x]	χ
[g]	γ
[o]	ο
[d]	δ
[p]	π
[e]	ε
[r]	ρ
[z]	ζ
[c]	ξ
[s]	σ
[h]	η
[t]	τ
[j]	ς
[u]	υ
[i]	ι
[f]	φ
[k]	κ
[q]	θ
[l]	ι
[y]	ψ
[m]	μ
[w]	ω

La asignación de acentos y signos de puntuación es como se ve en la Tabla 2

el fichero de configuración de las X, `/etc/X11/XF86Config-4`. Una de sus secciones está dedicada expresamente a configurar la extensión `xkb`, veamos qué dice en este ordenador:

```
Section "InputDevice"
Identifier "Generic Keyboard"
Driver "keyboard"
Option "CoreKeyboard"
Option "XkbRules" "xfree86"
Option "XkbModel" "pc105"
Option "XkbLayout" "es,us"
Option "XkbOptions" "grp:alt_shift_toggle"
EndSection
```

¿Qué tiene de llamativo el fragmento anterior? Que la combinación de las teclas `Alt` y mayúscula me permite alternar cíclicamente mi mapa de teclado español y un teclado estadounidense.

En `/etc/X11/xkb/README.config`, de muy interesante lectura, nos informan de que es posible alternar hasta cuatro mapas de teclado. ¿Funciona esta solución? Vamos a comprobarlo pulsando las teclas que tienen impresa la letra «ñ» y el signo de menor que, «<», y permutando los teclados. Con el mapa español la salida es «ñ <» como podíamos esperar; con el norteamericano es «; <». Pero si pulso la combinación otra vez no me devuelve al español, sino que obtengo «' «». ¿Qué está pasando aquí? Hay una forma de averiguarlo.

```
$ xprop -root | grep XKB
_XKB_RULES_NAMES_BACKUP(String)=2
"xfree86", "pc105", "es,us", 2
",", "grp:alt_shift_toggle"
_XKB_RULES_NAMES(String) =2
"xfree86", "pc105", "es,us,e1", 2
",,polytonic", 2
"altwin:menu,compose:rwin, 2
grp:alt_shift_toggle,lv3:lwin_switc"
```

Un nuevo agente inesperado interviene en el proceso. El applet «Indicador del teclado» de Gnome tiene un nombre engañoso; no es sólo indicativo sino que intercepta en memoria y sin modificar el fichero de configuración `XF86Config-4` las llamadas a `xkb` y nos permite, en un entorno Gnome y para el entorno `Gnome[1]`, definir de forma gráfica nuevas opciones y nuevos mapas (figura 6). Si un mapa de teclado

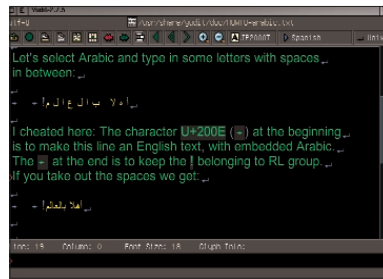


Figura 3: Árabe en yudit.

está disponible no hay más que seleccionarlo y ya podremos escribir con esa escritura. ¿Podemos generalizar esta solución al resto de los entornos? No, a menos que incorporemus la información de la salida anterior a la configuración de las X. Ahora estamos en condiciones de comprender lo que estaba ocurriendo: teníamos una tercera opción de teclado, el griego politónico (conste que ya lo sabía, para eso lo había configurado yo). `εὐρεκα!` ¡Ya podemos escribir en griego!

Pero hemos dicho politónico. Ah, puede que no sea tan sencillo: implica que funcione la composición de símbolos mediante el uso de las teclas muertas (o mudas, como nuestros acentos, que es necesario pulsarlos dos veces para que aparezcan solos). Y esta composición está vinculada al *locale* con el que estemos trabajando. En un muy interesante hilo en la lista de correo `linux-utf8` (*Locale for typing ancient Greek*, iniciado el 19 de febrero de 2005) Pablo Saratxaga nos da información muy pertinente: «Si usamos `LC_ALL=el_GR.UTF-8` podremos introducir con las teclas muertas todo el griego politónico, pero probablemente sólo un conjunto limitado de los acentos latinos. En cambio con `LC_ALL=en_US.UTF-8` podremos teclear gran cantidad de distintos acentos

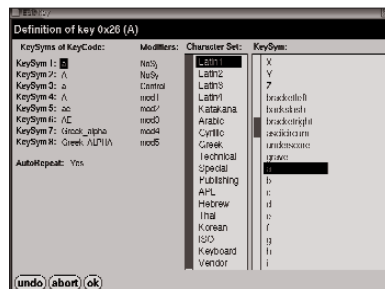


Figura 5: Configuración de la tecla «a» en `xkeycaps`.

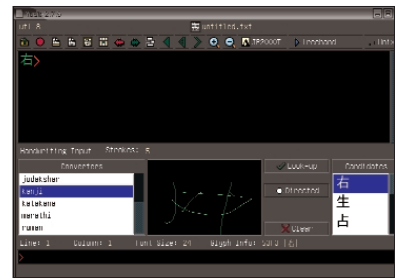


Figura 4: Yudit permite escribir a mano.

latinos, pero únicamente griego monotónico. Y con `LC_ALL=ja_JP.UTF-8` no será posible usar ninguna tecla muerta.» Es cuestión de probar: solo consigo á.

De entrada sopa de letras

ᾶᾷ. Bonito, ¿eh? ¿Cómo lo he hecho? Cambiando de enfoque. El applet de teclado de Gnome o los distintos mapas de `xkb` son soluciones limitadas. ¿Cómo introducir los miles de caracteres del chino? No necesitamos ni queremos estar cambiando de teclado continuamente, necesitamos un teclado que nos permita introducir de alguna manera todos los caracteres. ¿Existe esta solución? Como el lector ya sabe que mis preguntas son retóricas esperará una respuesta afirmativa; sí, los *métodos de entrada* (abreviados con IM, de Input Methods) se han creado para solucionar nuestro problema. Son mecanismos que permiten introducir caracteres de más de ocho bits (como los del chino) y realizar transliteraciones (introducir la representación fonética de una palabra y que el sis-

Tabla 2:Asignaciones de acentos y signos

Tecla	Acento o signo
<	espíritu áspero
>	espíritu suave
' (apóstrofo)	acento agudo
` (pulsar dos veces)	acento grave
~ (AltGr+ñ)	acento circunflejo
¨	diéresis
ı (AltGr+1)	iota suscrita
?	interrogación
;	punto alto

Se puede teclear los acentos y los espíritus en cualquier orden *antes* de la vocal, pero la iota suscrita siempre tiene que ser tecleada *después* de la vocal con acentos.

tema la transforme en el carácter de la escritura). Hay varios IM disponibles; de hecho Gnome (más exactamente GTK2) trae uno *de serie*, GTK+ IM. Otros son SCIM, UIM, IIMF... que deberán ser instalados aparte. Toda una sopa de letras para introducir letras y una nueva abreviatura, *m17n*, para un nuevo esfuerzo, la «multilingüalización» (sí, 19 letras). Por motivos que no es el caso explicar utilizo gedit para escribir estos artículos. ¿Qué ocurre si pulso el botón derecho sobre el texto que escribo? Que nos encontramos «Métodos de entrada» como una de las opciones del menú emergente. Seleccionando cirílico ya tenemos un teclado ruso. Pero el desafío está en el chino. ¿Cómo podemos escribir caracteres chinos con nuestro teclado español? Cada IM da su respuesta; elegimos SCIM (los siguientes paquetes funcionan sin problemas: *scim-gtk2-immodule*, *scim-uim*, *scim-chinese*, *scim-m17n*, *uim-gtk2.0*; yo instalaría también *m17n-env* pero no es necesario), más los añadidos UIM para SCIM y sus extensiones para que se acople con Gnome (añade a las opciones del botón derecho el método de entrada SCIM). Una vez instalado y seleccionado SCIM como método de entrada para las aplicaciones GTK, se llama con la combinación de teclas *Ctl-Espacio*. ¿Funciona? La figura 7 es el resultado del juego de mi hija de diez años escribiendo las letras de su nombre (por supuesto he eliminado los caracteres fácilmente reconocibles; dejo como ejercicio para el lector averiguar cómo se llama la niña). ¿Cómo se escribe en chino? Utilizaré *UIM-m17n-zh-py*. El proceso combina la transliteración y las sugerencias: el escritor teclea el sonido del carácter y el sistema sugiere los varios símbolos que pueden tener esa fonía

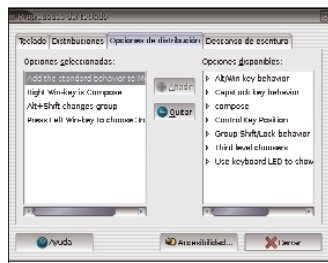


Figura 6: Configurando xkb con el Indicador del teclado.

para que se seleccione el deseado. Por ejemplo al teclear la sílaba «ca» se me ofrecen tres caracteres de la fuente a elegir.

Se nos quedan en el tintero muchos temas que no podemos tratar, al menos en este momento: diccionarios, recursos para el aprendizaje de alfabetos o de idiomas... Pero es hora de pensar en cerrar el artículo.

De aquí a China

Tras este espero que ilustrativo paseo por Grecia, Rusia y China debemos regresar al punto de partida, que será el de llegada: ¿qué tenía que hacer el profesor para que sus dos alumnas chinas se sintieran en casa?

Acabemos el artículo con una esquemática receta en diez puntos

- 1.- Crear dos *locales* chinos, *zh_TW.BIG5* y *zh_TW.UTF-8* (perdón por las implicaciones políticas posibles pero las niñas tienen que ser de algún sitio concreto y elijo Taiwán). Es preferible el uso de *utf-8*; de ese modo las herramientas de entrada permiten escribir otros alfabetos y es un requisito del IM preferido, pero aun un amplio conjunto de aplicaciones que sólo funcionan con los *locales* tradicionales.

- 2.- Reconfigurar *localepurge*, para que no borre los ficheros de estos nuevos *locales*.

- 3.- Instalar tipos de letra apropiados; sugiero el uso de tipos vectoriales (Type1 or TrueType) ricos en caracteres *unicode*: *bitstream-vera*, *arphic*; *Bitstream Cyberbit* es el mejor tipo *unicode* semilibre, y *Code2000* es *shareware*. Recuerde que para tipos instalados localmente es necesario refrescar la cache con *fc-cache*.

- 4.- Un conversor de codificaciones: *zh-autoconvert*.

- 5.- Ficheros de ayuda: *manpages-zh*.^[2]

- 6.- Diccionarios: *pydict*.

- 7.- Entorno de escritorio: actualmente uso Gnome. Gnome-terminal con *pango* funciona perfectamente con *locales utf-8*.

- 8.- Método de entrada. Para usuarios de *gtk/gnome*, *SCIM*.

- 9.- Para la enseñanza de la escritura de los caracteres chinos: *hanzim*.

- 10.- Instalar los archivos localizados: *openoffice.org-help-zh-tw*, *kde-i18n-zhtw*, *mozilla-locale-zh-tw*...Y el resto de los añadidos habituales: *fortune-zh*, *jugos localizados*...

Y en el próximo número...

¿Qué tal hablar sobre las adaptaciones y aplicaciones destinadas a la atención a la diversidad y a los discapacitados de todo tipo? Éste, el de la accesibilidad, es un tema que se demanda mucho, por el que algunos se resisten a introducirse en el software libre y se desconocen las soluciones libres existentes. ■

NOTAS

[1]Gnome tiene su propio mecanismo de buceo (información obtenida en la lista de correo *linux-utf8*, hilo *Thank God for Yudit*, iniciado el 14 de febrero de 2005)

```
$ gconftool-2 -R /desktop/gnome/
peripherals/keyboard/xkb layouts =
[es,us,el polytonic]
model = pco105 overrideSettings =false
options = [altwin altwin:menu,
compose compose:rwinn.grp
grp:alt_shift_toggle,lv3
lv3:lwin_switch]
update_handlers = []
```

[2]man presenta un problema; he leído que *groff* puede producir *utf-8*, pero no así *troff*. No he conseguido generar páginas de manual apropiadas, algo debo hacer mal.

EL AUTOR

Juan Rafael Fernández García es profesor de educación secundaria y tiene una larga experiencia en la traducción y documentación del software libre. Ha sido coordinador de uno de los Centros que participan en la experiencia andaluza de integrar las TIC en la educación y actualmente trabaja como asesor de formación del profesorado.