



Bisoños Usuarios de GNU/Linux de Mallorca y Alrededores | Bergantells Usuaris de GNU/Linux de Mallorca i Afegitons

Internacionalizando que es gerundio (8294 lectures)

Per Daniel Rodriguez, [DaniRC](http://www.ibiza-beach.com/) (<http://www.ibiza-beach.com/>)

Creado el 27/01/2003 17:53 modificado el 27/01/2003 17:53

Traducir es una de esas tareas que uno puede llevar a cabo en el mundo Linux sin necesidad de grandes conocimientos técnicos.

Como aún así las herramientas de traducción e internacionalización son unas grandes desconocidas voy a tratar de acercarnos al **gettext** y en concreto al **pygettext**. La herramienta que usaremos para internacionalizar nuestro código **Python**.

¿ Cómo internacionalizo mis programas ?

Ejemplo hola.py internacionalizado:

Si traduzco a la vez que escribo no hará falta el `def _(mensaje)` pero si primero programo y luego traduzco esto hará que todo funcione en espera de que llegue el diccionario.

Programa hola.py

```
def _(mensaje):
```

```
    return mensaje
```

```
print _("Hola mundo")
```

Hecho esto paso por el `pygettext`

```
pygettext --output=hola.po hola.py
```

En el `hola.po` hago las traducciones oportunas. Por ejemplo para el idioma catalán:

```
# HOLA MUNDO en catalán
```

BULMA: Internacionalizando que es gerundio



```
# Copyright (C) 2003 Dani
# Dani <daniATATATbulma>, 2003.
#
msgid ""
msgstr ""
"Project-Id-Version: Hola mundo 0.1\n"
"POT-Creation-Date: Sun Jan 26 18:49:09 2003\n"
"PO-Revision-Date: YEAR-MO-DA HO:MI+ZONE\n"
"Last-Translator: DANI <daniATATATbulma>\n"
"Language-Team: Catalan <ca@li.org>\n"
"MIME-Version: 1.0\n"
"Content-Type: text/plain; charset=iso-8859-1\n"
"Content-Transfer-Encoding: 8bit\n"
"Generated-By: pygettext.py 1.3\n"
```

```
msgid "Hola mundo"
msgstr "Hola mon"
```

y lo guardo con el nombre hola.po

Ahora necesito el msgfmt y hago esto:

```
msgfmt --output=hola.mo hola.po
```

El fichero .mo es el fichero que se usa para el diccionario de traducciones. Es un fichero "compilado".

Ahora llevo el hola.mo al directorio:

```
/usr/share/locale/ca/LC_MESSAGES/
```

porque alli es donde se buscan las traducciones por defecto para el idioma ca_ES

Ahora que ya tengo instalado el diccionario de traducciones tengo que cambiar el programa hola.py para que coja los textos del diccionario cuando sea necesario.



```
hola.py

#Añado esto

import gettext

#Aqui debe ir el nombre del modulo o del programa ... que es #el nombre que le hemos dado al .mo

gettext.install("hola")

#Quito esto que es lo que usaba antes de tener lista la traduccion.

#def _(mensaje):

# return mensaje

print _("Hola mundo")
```

Las pruebas

Ahora puedo probarlo haciendo un

```
$ export LANG="ca_ES"
```

```
$ python hola.py
```

Hola mon

y luego un

```
$export LANG="es_ES"
```

```
$python hola.py
```

Hola mundo

¿ Cómo traduzco un programa ya creado ?

Nota: Suponemos que el creador supuso que su programa iba a ser internacionalizado ;-)

Con Debian, lo mas facil es hacer un

```
apt-get sources programa_favorito
```



Esto te dejara las fuentes en el directorio local ... alli encontraras posiblemente un directorio po te metes alli dentro y creas tu propio ca_ES.po en base a algun fichero que haya por ahí con la traduccion al castellano es_ES por ejemplo.

Ok. Traduces todo lo que puedas y lo dejas ahi dentro.

Luego compilas e instalas el paquete al estilo debian, Segun <http://bulma.net/body.phtml?nIdNoticia=1609>

¿Cómo compilo el código fuente?

Dentro del directorio que te ha generado el apt-get-source haces:

```
fakeroot dpkg-buildpackage
```

El fakeroot te permite compilar y generar los .deb sin necesidad de ser root. Al finalizar verás que en el directorio inmediato superior uno o varios .deb. Ya podrás hacer un dpkg -i

Si todo fue bien el proceso de instalacion ejecute el msgfmt por tí .. si no es el caso ... haz tu mismo el msgfmt sobre tu fichero .po y metelo a mano en el LC_MESSAGES de tu idioma.

Ahora ya deberias tener tu aplicacion traducida :-)

Nota: Por si acaso haz algunas pruebas con gaseosa ... no sea que traduzcas 400 folios y luego falle y te acuerdes de todos mis parientes ;-)
¿vale?

Esto vale para programas en C, en Python y para aquellos que no son de KDE o Qt porque las KDE y las Qt usan el QtLinguist que en mi opinion es mucho menos practico.

Buena suerte ... y a traducir que esta tirado y todos podemos hacerlo ;-)

E-mail del autor: danircJUBILANDOSEbulma.net

Podrás encontrar este artículo e información adicional en: <http://bulma.net/body.phtml?nIdNoticia=1675>