

Programmeur, traducteur, lecteur, l'écosystème de la traduction

Jean-Philippe Guérard

Traduc.org

14 septembre 2008

Introduction

- Traduc.org, c'est quoi ?
- Programmeur, traducteur, lecteur.
- Un sombre dessein...

Pourquoi traduire ?

- Un logiciel uniquement en anglais aura un public restreint.
- Pour rendre les logiciels simples à utiliser... et à aimer.
- Pour les débutants... et les autres.

Le programmeur et gettext

- Il lui suffit d'annoter son programme.
- Il doit faire attention à la rédaction des textes.
- Éviter le saucissonnage et les ambiguïtés.
- Prenez le temps de lire la documentation de Gettext.

Un petit programme bien inutile

```
#!/usr/bin/env python
# -*- coding: UTF-8 -*-

print u'Hello, World.'
```

On peut exécuter ce script, qui donne le résultat attendu :

```
$ ./essai.py  
Hello, World.
```

Les lourds secrets de la traduction

```
#!/usr/bin/env python
# -*- coding: UTF-8 -*-

import gettext
gettext.bindtextdomain('essai', '.')
gettext.textdomain('essai')
_ = gettext.gettext

# TRADUCTEURS : ce message ne doit pas
# dépasser 72 caractères
print _(u'Hello, World.')
```

Un dernier effort

Il ne reste qu'à extraire le texte à traduire :

```
xgettext -L python -o essai.pot \  
  --add-comments='TRADUCTEURS :' \  
  essai.py
```

Roulements de tambours...

Le résultat sera la base du travail du traducteur :

```
#. TRADUCTEURS : ce message ne doit pas  
#. dépasser 72 caractères  
#: essai.py:13  
msgid "Hello, World."  
msgstr ""
```

Ce fichier, renommé `essai.po` sera envoyé aux traducteurs.

Que faire avec ce machin ?

- Trouver des traducteurs (et vice-versa).
- Intégrer les traducteurs (si, si, vraiment).
- Motiver et conserver les traducteurs.
- Organiser les traductions.
- Comment contrôler la qualité du résultat ?

Le traducteur et les renards véloce

- Comprendre, puis reformuler.
- On ne peut traduire ce que l'on ne comprend pas.
- Un travail de recherche.
- Comment expliqueriez ça en français ?
- Produire un texte simple et qui sera instinctivement compris.

Résister à l'envie de coller à l'original

- Une force puissante.
- L'objectif est d'être compris.
- Attention à la perspective du lecteur !
- L'objectif est d'être utile, pas de reprendre fidèlement les erreurs.
- Signaler les erreurs à l'auteur.

Quelques conseils (1)

- Traduisez des logiciels que vous utilisez.
- Une fois le logiciel traduit, restez.
- Attachez-vous à traduire les documents associés.
- Traduisez des documents qui vous intéressent.

Quelques conseils (2)

- Toujours se relire soi-même.
- Toujours demander en cas de doute.
- Ne jamais, jamais traduire sans comprendre.
- Toujours signer son travail.
- Créer et entretenir un lexique. Le publier.

Un petit exemple...

On ajoute *simplement* la chaîne manquante au-dessous du texte original :

```
#. TRADUCTEURS : ce message ne doit pas  
#. dépasser 72 caractères  
#: essai.py:13  
msgid "Hello, World."  
msgstr "Bonjour, Monde."
```

Un petit exemple...

Reste à installer la traduction :

```
mkdir -p fr/LC_MESSAGES  
msgfmt -c -o fr/LC_MESSAGES/essai.mo essai.po
```

On peut ensuite essayer le logiciel :

```
$ ./essai.py  
Bonjour, Monde.
```

Relire

- Utiliser un correcteur orthographique.
- Valider le fichier « .po ».
- Lancer le logiciel traduit et essayer de retrouver tous les textes dans leur contexte original.
- Seule la relecture permet de savoir si la traduction est bonne.

Les projets de traduction (1)

- Manque chronique de traducteurs.
- L'effort est mis sur les traductions plutôt que sur les relectures.
- Chaque projet a un domaine d'intervention limité (distribution, environnement de bureau, documents spécifiques).

Les projets de traduction (2)

- Manque de visibilité.
- Chacun réinvente la poudre de son côté.
- Un outil commun : `glossaire.traduc.org`.
- Il manque un méta-projet international.

Monter son projet de traduction

- Regardez d'abord les projets existants.
- Quelques pages de wiki suffisent.
- Faites-vous connaître.
- Soyez prêt à tout faire vous-même, au moins au début.

Le lecteur et les traductions

- Il a tendance à voir les logiciels libres comme un produit du commerce.
- Frustration et insatisfaction devant l'absence de traduction ou les mauvaises traductions.
- Il a du mal à savoir qui fait quoi.

Le lecteur : consommateur passif ?

- Philosophie du libre : l'utilisateur est responsable de la qualité du logiciel. Il doit signaler les problèmes.
- L'utilisateur est responsable de la qualité des traductions.
- Un rôle accru des utilisateurs permettrait d'améliorer la qualité.
- Peu de systèmes de retour utilisateur accessibles.
- Quelles solutions ? (entrées de menus, formulaires en français, et cætera.)

On se réveille !

- Lecteurs : ne soyez pas consommateurs passifs, impliquez-vous.
- Programmeurs : intégrez les traducteurs à votre processus de développement.
- Traducteurs et programmeurs : donnez aux lecteurs les moyens de vous aider !

Vos questions