

Educadores traduzindo software livre educacional

Rainer Krüger – Pandorga GNU/Linux

RESUMO: Este artigo busca mostrar que traduzir um software livre não é tão complicado quanto parece. É um estímulo aos educadores colaborarem com o desenvolvimento dos sistemas que usam com seus alunos. Será mostrado quais são os passos para se traduzir um software, incluindo alguns conceitos como internacionalização, vocabulário padrão e ferramentas que facilitam o trabalho.

PALAVRAS-CHAVE: Educação. Tecnologia. Linguagem.

Introdução

Este artigo, apesar do título parecer técnico, é um roteiro que mostra como pessoas sem conhecimentos de programação podem participar da tradução de software livre. Ele é direcionado à educadores, mas pode ser usado por qualquer pessoa em qualquer área, mudando apenas as fontes de pesquisa.

Meu foco em educação é parte do dia-a-dia, mantenho o Pandorga GNU/Linux[1] que é uma distribuição Linux para o ensino fundamental, também participo do Grupo de Tradução de Software Livre Educacional, o SLEducacional[2]. Mas considero o assunto abordado relevante pois a educação deve ter uma especial atenção na tradução de seus programas, estamos lidando diretamente com a fase de aprendizagem das crianças, e em especial no ensino infantil e fundamental, com a própria aquisição da linguagem. Usar um programa em língua estrangeira é ruim, mas pior é usar um programa em português com termos e construções inadequadas. Isto pode influenciar diretamente seu desenvolvimento da língua.

É comum que desenvolvedores de sistemas, como eu, não sejam experts em português, além de muitas vezes não terem conhecimento profundo das didáticas em aula. Logo, é mais indicado que um educador realize a construção dos textos de um programa. Como muitos dos programas usados em Software Livre são desenvolvidos no exterior, cabe a nós traduzir e adaptar partes do mesmo, tarefa que pode ser facilmente feita por um educador. Duvidam? Vou lhe mostrar agora...

Traduzir

Traduzir é o ato de transformar os termos e textos de uma língua para outra, no contexto de software, o ato de transpor as mensagens existentes num determinado programa em língua estrangeira para a brasileira, mantendo seu significado e características funcionais do programa.

Traduzir é, além de transformar a língua, o ato de interpretar, manter o significado ao contexto em questão, e muitas vezes, adaptar termos e construções para a realidade local.

Em algumas situações devemos ainda verificar se o conhecimento disponível no software é realmente aplicado à nossa realidade, como programas voltados para línguas, política, história ou geografia. O tradutor deve dominar o assunto para que os textos não fiquem confusos ou incorretos, devemos saber o que aquele programa deseja passar para o aluno, que tipo de atividade será desenvolvida com este programa.

Incentivar os educadores, e demais profissionais com conhecimento na área de educação a traduzir, isso garante maior qualidade nos programas e na educação. Eles podem também analisar formas para melhor empregar o software em aulas/laboratórios de aprendizagem, incentivando estes profissionais a participar do processo de desenvolvimento dos programas.

Ao contrário que se pensa, para se traduzir um programa não é necessário alto conhecimento da língua estrangeira, uma vez que os textos e mensagens costumam ser pequenos e diretos, basta entender seu significado para realizar a tradução.

Software Livre e Educação

Software é um programa ou conjunto de programas para computador que realiza determinada tarefa, podemos chamar de software o sistema operacional e os programas que usamos no computador. Software Livre é a licença de uso do software, ela define que qualquer pessoa é livre para usá-lo para qualquer propósito em quantos computadores desejar, e ainda pode estudá-lo e alterá-lo como quiser. Ele não define que o software seja obrigatoriamente gratuito, mesmo que a sua grande maioria seja.

Para realizar a tradução de um software, é necessário ter acesso ao seu código-fonte, a receita de como ele foi feito. Num software proprietário isto fica restrito aos desenvolvedores, por questões técnicas e jurídicas. Tecnicamente, quando adquirimos um software proprietário, temos somente seu executável, não podemos modificá-lo ou traduzi-lo. Em questões jurídicas, mesmo que de alguma forma tenhamos acesso ao código-fonte do sistema, não poderemos alterá-lo e nem mesmo redistribuí-lo.

A filosofia do Software Livre é muito próxima a filosofia educacional, onde a colaboração e a transferência de conhecimentos são peças-chaves para seu sucesso. A transferência de conhecimento é a premissa básica da educação, enquanto que no Software livre é fortemente

incentivada para que o movimento se permaneça e as pessoas saibam usá-lo. Da mesma forma, a colaboração é a premissa do software livre, já a educação usa a colaboração para a construção do conhecimento. Usar o computador para fins educativos de forma limitada, é o mesmo que dar um livro ao aluno e não permiti-lo entrar na biblioteca para estudar mais. Segundo o estudo de Ana Cristina Matte [3], “[o Linux] trata-se de um sistema tensivo convergente: quanto mais eu uso, mais eu sei sobre o próprio sistema”.

O que precisamos saber para traduzir

O computador não entende português, inglês ou qualquer outra língua falada, ele possui uma língua própria, construída de forma lógica e seqüencial. Para se construir um programa devemos dar instruções exatas das tarefas que desejamos para que o computador realize em ordem também exata. Usaremos uma linguagem de programação (existem diversas) para escrever nossos programas, depois um compilador irá transformá-lo em linguagem de computador. A nossa escrita é o código-fonte, enquanto que seu resultado é o executável, ou binário.

As mensagens e textos existentes no programa ficam no código-fonte e podem estar organizadas de duas formas: a forma tradicional insere as mensagens junto à lógica de programação, complicando sua tradução e não permitindo a internacionalização. Já a segunda forma, usa um arquivo separado para definir as mensagens, dentro da lógica fica chaves que buscam estes textos, esta forma permite que o programa esteja traduzido para várias línguas e que facilita muito sua tradução.

Esta segunda forma é conhecida por `i18n`, abreviatura de internationalization (i mais 18 letras e n), que representa a possibilidade de se manter mais de uma língua no mesmo programa. Para se traduzir um programa da primeira forma citada acima, deve-se traduzi-lo integralmente e manter uma versão local do sistema ou internacionalizá-lo, opção mais indicada.

Após traduzido, o sistema deve ser compilado e distribuído, num projeto já em andamento envia-se o arquivo de tradução para o projeto e eles irão fazer o restante.

i18n

Agora que sabemos como funciona um programa, iremos trabalhar na tradução em si e saber como a `i18n` funciona.

Os sistemas operacionais possuem uma configuração chamada Locale que define características regionais do sistema. Estão na lista de características a língua falada e formatação de números e moedas. Cada região geográfica/social/política possui uma Locale ou l10n (Localization), a brasileira é pt-BR. Primeiro nós temos a língua falada, pt (português) depois a região, BR, ou seja, somente pt é o português de Portugal. Outros exemplos são en (inglês), en-US (Estados Unidos), es (espanhol), es-AR (Argentina), es-CL (Chile) e assim por diante. Pode-se consultar mais informações e a lista completa em <http://www.w3.org/International/>.

Quando um programa que utiliza l10n é executado, ele procura no sistema operacional qual é a Locale padrão, e carrega a tradução correspondente. Caso o programa não traduza para a Locale, ele é executado na língua padrão. O que faremos ao traduzir é adicionar a regionalização pt-BR para o nosso programa.

Na maioria das linguagens de programação a internacionalização fica num arquivo com extensão po (Portable Object) conhecidos por Modelo PO. Nestes arquivos ficam as mensagens originais e sua tradução, com um possível comentário sobre a tradução. Muitas vezes o trabalho de tradução já foi iniciado bastando apenas complementar, em outros casos deve-se criar o arquivo po e realizar toda sua tradução.

O exemplo a seguir é do Kalzium, um programa de química do projeto KDE, seu arquivo po está disponível no endereço http://l10n.kde.org/stats/gui/stable/team/pt_BR/kdeedu/ :

```
#: detailinfodlg.cpp:71
msgid "Goes to the previous element"
msgstr "Vai para o elemento anterior"

#: detailinfodlg.cpp:72
msgid "Goes to the next element"
msgstr "Vai para o próximo elemento"

#: somwidget_impl.cpp:142 somwidget_impl.cpp:156
msgid ""
"_: For example: Carbon (300K)\n"
"%1 (%2)"
msgstr "%1 (%2)"
```

Código-fonte 1: Arquivo kalzium.po

A primeira linha com (#) indica um comentário, será ignorado pela linguagem de programação, neste caso informa onde a mensagem está sendo usada no programa, arquivo detailinfodlg.cpp linha 71. Na segunda linha temos o termo em inglês e na linha seguinte temos o termo em português, isto se repete para o próximo termo a partir da linha 5. No último exemplo está sendo usado %1 e %2 na frase, isto significa que estes valores serão substituídos pelo sistema, como

(<http://gtranslator.sourceforge.net/>). Para as plataformas Linux e Windows existe ainda o Poedit.

De forma geral estas ferramentas possuem uma janela com o termo original e sua tradução, podem possuir também pesquisa no VP e informações sobre os termos ainda não traduzidos.

É essencial ao tradutor conhecer os sistemas SVN e CVS, ambos controladores de versão de arquivos. Este sistema possibilita que vários tradutores ou desenvolvedores trabalhem no mesmo projeto. Através de uma ferramenta cliente, você irá baixar a versão mais nova do arquivo que deseja alterar sempre que for usá-lo, quando salvar, ele é enviado diretamente ao servidor do projeto, garantindo que se outra pessoa for editá-lo, use a versão mais nova também. Ele possui também um sistema de registros onde você deve deixar uma mensagem sobre as alterações, e o próprio sistema é capaz de analisar o que foi modificado, gerando logs de alterações. No endereço a seguir pode-se encontrar mais informações sobre o SVN do KDE: <http://br.kde.org/index.php?title=Configurar_o_Subversion>.

Além de traduzir

Caso você ainda não se sinta a vontade em traduzir programas, existem uma grande variedade de formas para se colaborar, traduzir documentos e manuais pode ser a primeira forma. Tão importante quanto o software é sua documentação e ajuda.

Pode-se também auxiliar outros tradutores e desenvolvedores em fóruns e listas de discussão.

Existe também o avaliador ou revisor que é responsável por verificar se as traduções estão corretas. Você pode enviar ao projeto erros encontrados durante a utilização do sistema, é uma forma muito importante de colaborar.

Existem ainda grupos, como o SLEducacional, onde a participação de um educador é muito importante, seja traduzindo, documentando, avaliando, pesquisando programas, colaborando na análise da qualidade dos programas, trabalhando na criação gráfica, elaboração de planos de aulas e muitas outras possibilidades. Basta querer colaborar que você encontrará uma forma de ajudar.

A finalidade principal do software livre é dar às pessoas o acesso à tecnologia, de forma justa e livre. Para este mundo, quanto mais os usuários do sistema participar do seu desenvolvimento, melhor, mais qualidade e liberdade haverá. No software livre não existe concorrente, existe alternativas.

Conclusão

Espero neste artigo ter mostrado que não é difícil traduzir um software livre, basta conhecer um pouco a língua estrangeira, adquirir o arquivo de tradução do projeto, editá-lo e enviá-lo novamente aos desenvolvedores. Para isto, temos ferramentas como o KBabel ou o GTranslator que nos auxiliam na tradução, além dos sistemas SVN e CVS.

Busque participar dos projetos em software livre, você irá contribuir com a qualidade e a diversidade de sistemas voltados para a educação, fazendo da educação um momento melhor, mais divertido e mais amplo.

Referências

[1] Pandorga GNU/Linux – <http://pandorga.rkruger.com.br>

[2] SLEducacional – <http://seducacional.org>

[3] *Ana Cristina Fricke Matte -Tuxsila: quem disse que linux é difícil?*
<http://www.textolivre.net/revista/index.php/TextoLivre/article/view/5>

Outras referências:

Projeto educacional do KDE: <<http://edu.kde.org/>>

Traduções do KDE: <http://l10n.kde.org/team-infos.php?teamcode=pt_BR>

Traduções do GNOME: <<http://br.gnome.org/bin/view/GNOMEBR/Traducao>>

Vocabulário padrão brasileiro: <<http://vp.godoy.homeip.net/>>

Informações para iniciar a tradução de Potfiles do KDE:

<<http://br.tldp.org/projetos/kde/iniciante.html>>