

An In-Context and Collaborative Software Localisation Model: Demonstration

Amel FRAISSE Christian BOITET Valérie BELLYNCK

LABORATOIRE LIG, Université Joseph Fourier, 41 rue des Mathématiques, 38041 Grenoble, France

Amel.fraisse@gmail.com, christian.Boitet@imag.fr,

Valerie.Bellynck@imag.fr

ABSTRACT

We propose a demonstration of our *in context* and collaborative software localisation model. It involves volunteer localisers and end users in the localisation process via an efficient and dynamic workflow: while using an application (in context), users knowing the source language of the application (often but not always English) can modify strings of the user interface presented by the application in their current context. The implementation of that approach to localisation requires the integration of a collaborative platform. That leads to a new tripartite localisation workflow. We have experimented with our approach on Notepad++. A demonstration video is proposed as a supplementary material.

KEYWORDS: Software localisation, machine translation, translation memories, incremental and collaborative translation, in context localisation, collaborative localisation

Un modèle en-contexte et coopératif pour la localisation de logiciels : Démonstration

RESUME

Nous proposons une nouvelle approche qui permet la localisation en contexte et collaborative de la plupart des logiciels à source ouvert. Notre approche fait participer des bénévoles et les utilisateurs finals au processus de localisation via une organisation du travail efficace et dynamique: en même temps qu'ils utilisent une application ("en contexte"), les utilisateurs connaissant la langue source du logiciel (souvent mais pas toujours l'anglais) peuvent modifier des chaînes de l'interface utilisateur présentées par l'application dans leur contexte courant. L'implémentation de cette approche de la localisation requiert l'intégration d'une plate-forme collaborative. Cela mène à une nouvelle organisation du travail tripartite. Nous avons expérimenté et validé notre approche sur Notepad++. Une démonstration sera présenté.

MOTS-CLES: localisation de logiciels, traduction automatique, mémoire de traductions, traduction incrémentale et collaborative, localisation en contexte, localisation collaborative.

1 Introduction

Currently, the translation of technical documents as well as user interface strings is entrusted only to professional translators. In practice, localisation project managers¹ send original versions of the files to be localised to several professional translators. Each translator translates and sends the translated versions to the localisation project manager. It seems impossible to continue in this way for most under-resourced languages, for reasons of cost, and quite often because of the scarcity or even lack of professional translators (costs increase while quality and market size decrease).

On the other hand, free software such as that produced by Mozilla (Mozilla, 2009) is translated by volunteer co-developers into many (more than 70) languages, in some cases more languages than commercial software. The localisation process is based on the contribution of volunteers (Vo-Trung, 2004), (Tong, 1987), (Lafourcade, 1991, 1996). Another situation (different from the translation of technical documentation) is that of occasional volunteer translators, who contribute without an organic connection to the project (Linux, 2005). Hence, it is possible to obtain high quality translations of documents that may be over a hundred pages long (such as articles of Wikipedia, or texts of Amnesty International and Pax Humana).

Another problem of the classical localisation process is that strings of the interface are often translated out of context. Hence, the choice of the appropriate translation is not always possible due to lack of context, and in such cases even a professional translator cannot produce a perfect translation.

As proposed in (Boitet, 2001), one solution to this problem is to involve end users with a knowledge of the source language (often but not always English) and who, during the use of software products, translate or improve “pretranslations” produced by machine translation (MT) or translation memory (TM) systems.

2 The new in context and collaborative localisation model

2.1 Basic principles

2.1.1 Involving volunteer translators and end users in the localisation process

As said above, localisation seems impossible for most under-resourced languages for reasons of cost, and scarcity or even lack of professional translators.

Our solution aims at involving non-professional translators such as volunteer localisers and above all end users. These groups have the capacity to participate effectively, since they have a better knowledge of the target language (generally their native language) and of the context of use of the application. In order to motivate this type of translators and to give them a better knowledge about the use context of UI (user interface) strings, localisation should be carried out while using the application.

¹ Localisation project managers: software publisher in the case of commercial software, and a community of volunteer localisers in the case of open source software.

2.1.2 From discontinuous, coordinated and out-of-context localisation to continuous, uncoordinated and in context localisation

The basic concept of our model is to renounce the idea of perfect translation, and to publish rough translations with a variable quality, which will be improved incrementally during the use of the application. Therefore, the translation process will be ongoing and improve continuously. This solves the problem of time and delays, since users do not have to wait for the final localised version in their language. They can download, at any time, a partially localised or non-localised version of the application.

Similarly, the localisation project manager may first publish a partially localised version that will be progressively localised through use, leading, eventually, to a completely localised version. Hence, the new process permits the incremental increase of both quality and quantity.

The same principle is already applied by many translation communities. The best known is the Wikipedia Community: content is added and translated continuously by contributors. Our guiding idea is to adapt this principle to the software localisation field.

2.2 Adaptation to software localisation

The localisation project manager should be allowed to ask professional translators and reviewers to translate the crucial parts of the software. Hence, our localisation model has to be applicable individually or collaboratively. The user has the choice to localise the application *locally*, without any exchange with other users, or to localise it *collaboratively*.

2.2.1 First Scenario: in context localisation through interaction with the collaborative platform

During the in context localisation, user can interact with the collaborative platform (SECTra_w), to get and submit translations. As shown in FIGURE 1, when the user right-clicks on any string within the interface, an edition pop-up related to the collaborative platform appears, and the user can enter a new translation, or choose one of the proposed translations. Clicking on the "localize" button sends the user translation to the collaborative platform, and the user interface is updated in real time.

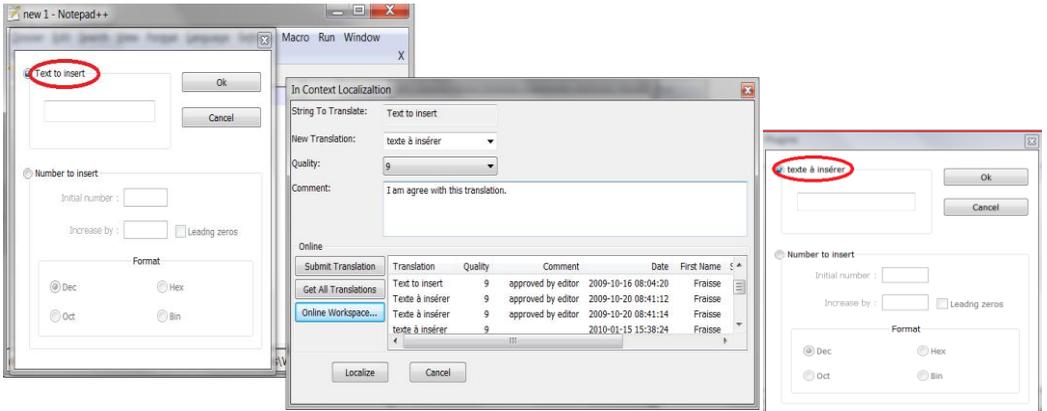


FIGURE 1 – In context localisation of the string "Text to insert" through interaction with the collaborative platform.

2.2.2 Second scenario: localising directly on the collaborative platform

This second scenario allows user to localize directly on SECTra_w (FIGURE 2). When the edition pop-up is displayed, the user clicks on the "Online workspace" button and is redirected to the page on the collaborative platform containing the string that has been chosen for translation. Then, the user enters a new translation, or chooses one of the proposed translations. When, s/he returns to the application (in the same context s/he left it), the interface has been updated.

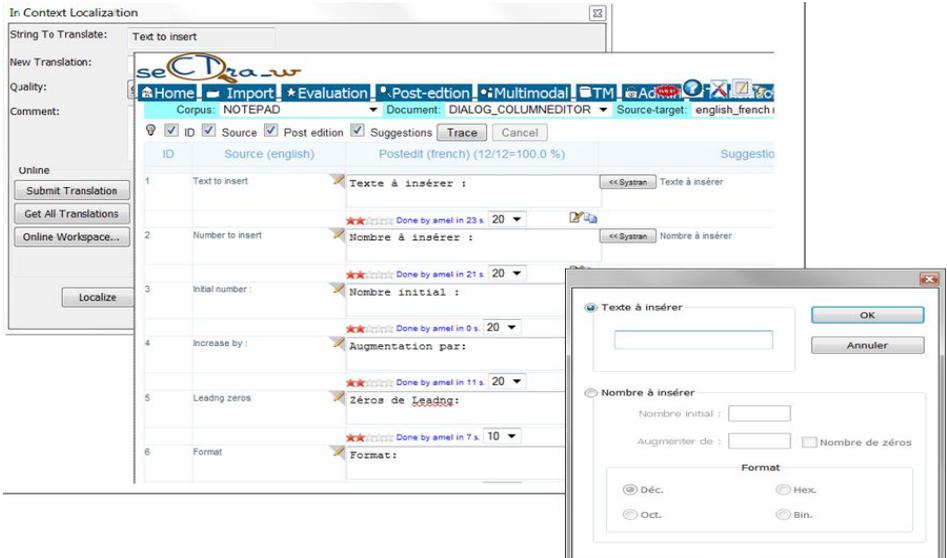


FIGURE 2 – Localising directly on the collaborative platform.

References

- Bey, Y., Kageura, K. and Boitet, C. (2008). BEYTrans: A Wiki-based Environment for Helping Online Volunteer Translators. *Topics in Language Resources for Translation and Localisation*, Yuste Rodrigo, 2008, 135–150.
- Blanchon, H., Boitet, C. and Huynh, C. (2009). A Web Service Enabling Gradable Post-edition of Pre-translations Produced by Existing Translation Tools: Practical Use to Provide High-quality Translation of an Online Encyclopedia. *MT Summit XII*, Ottawa, Ontario, Canada, 26-30 August 2009, 8 p.
- Boitet, C. (2001). Four technical and organizational keys for handling more languages and improving quality (on demand). In *Proc. Machine Translation System*, IAMT 8 p., Santiago de Compostela, September 2001.
- Fraisse, A., Boitet, C., Blanchon, H. and Belynyck, V. (2009). Localisation interne et en contexte des logiciels commerciaux et libres. *MajecSTIC Manifestation des jeunes chercheurs en sciences et technologies de l'information et de la communication*, Avignon, France, 16-18 November 2009.
- Huynh, C., Boitet, C., and Balnchon, H. (2008). SECTra_w.1: an online collaborative system for evaluating, post-editing and presenting MT translation corpora. *LREC 2008: 6th Language Resources and Evaluation Conference*, Marrakech, Morocco, 26-30 May 2008, 6 p.
- Lafourcade, M. and Sérasset, G. (1996). Apple Technology Integration. A Web dictionary server as a practical example. In *Mac Tech magazine*, 12/7, 1996, pp. 25.
- Lafourcade, M. (1991). ODILE-2, un outil pour traducteurs occasionnels sur Macintosh. In *Presses de l'université de Québec*, Université de Montréal, AUPELF-UREF ed., 1991, pp. 95-108.
- Linux (2005). Linux documentation translation [online], available at: <http://traduc.org/> [last accessed 12 May 2011].
- Mozilla Localisation Project (2009), available at <http://www-archive.mozilla.org/projects/l10n/> [last accessed 12 May 2011].
- Tong, L.C. (1987). The Engineering of a translator workstation. In *Computers and Translation*, pp. 263-273.
- Vo-Trung, H. (2004). *Méthodes et outils pour utilisateurs, développeurs et traducteurs de logiciels en contexte multilingue*, PhD thesis.