Tamil Localisation Process – A case study

Kengatharaiyer Sarveswaran sarvesk@uom.lk

Gihan Dias gihan@uom.lk

Department of Computer Science and Engineering, Faculty of Engineering,
University of Moratuwa, Sri Lanka

Abstract

Localisation has become an active area in computer field and many organisations and individuals are localising software into their preferred languages. Different people use different localisation processes to do localisation. We cannot see much difference in the existing localisation processes. There can exist more than one localisation for a language. This difference may occur if localisers follow different glossaries and style guides. Many software are localised to Tamil language too. The aim of this paper is to describe a Tamil localisation process that is followed in Sri Lanka.

Keywords : Tamil , Localisation, Localisation Process, Locale

Background

Localisation is the process of modifying products or services to account for differences in distinct markets including language and cultural differences [3]. In software domain, converting Graphical User Interfaces (GUI) to a language while considering local policies and cultural factors can be referred to as software localisation. With the boom of Free and Open Source Software (FOSS) many organisations and individuals have started to localise a number of FOSS operating systems and Application software [2]. FOSS offers a great deal of freedom in contributing to it so that many organisations and individuals come forward from all around the world to localise FOSS. And the number of languages a FOSS support also has become marketing factor now. Not only the FOSS organisations but also the proprietary software industries also do localise their software mainly to capture the markets.

The language and other factors like date and time format, measurements, number formats, currency are collectivity referred to as locale, viz EN, en, en_US etc. A new locale is created when software is localised into a new language. Some organisations release their original software and they release the language packs separately, on the other hand some software are released directly as localised versions. These localised versions of software and languages packs are identified using locales. There are different naming conventions followed to name a locale. Mostly the locale name contains two-letter notation according to the ISO 639-1 standard [11]. However when a single language is used in two places with different parameters mentioned above, then in addition to the ISO 639-1 language code, the ISO 3166 version of country codes [7] are also joined using an underscore or a hyphen. en_US and en_UK, en-US and en-UK are some examples for this naming convention.

Different organisations follow different processes to localise software. However the main steps involving in localisation include extracting the source files, translation, testing and packaging [8][6]. Since anyone can do the localisation and also many software need to be localised there are chances for lot of confusions and inconsistencies mainly in the translations and the way the words are translated. To overcome these issues few standards such as Glossaries and localisation style guides are used during the translation phase. Glossaries are well known documents which have alphabetical list of terms in a particular domain of knowledge with the translation of those terms. GUI may consist of many elements such as menus, dialog boxes, buttons, user messages etc. There are different factors that should be considered when translating these elements. Different organisations follow different policies

in translating these elements. Style guides represent how these elements should be translated. Not only these elements but also the things like how to translate acronyms, how to assign access keys, what tense to be used and where these tenses to be used, what plural rules to be used are specified in Style guide [5].

Technologically there are many tools and techniques that have been introduced to ease the tasks in each phase of localisation process. Mainly in the tedious translation phase many tools have been used based on the type of source files. For example to translate Portable Object (PO) files, the tools like POEdit, Pootle can be used [13]. During the translation the localisers may come across the terms that may repeat and also the terms that have been translated already for different software. The technique called translation memory makes it possible to reuse such terms. Many Computer Aided Translation tools support for these techniques to automate the translation to some extent. Since the localisation is not just dictionary translation, translation tools may not be useful in this context.

Localisation efforts in Sri Lanka

Localisation is very important in developing countries and it gives many benefits [1]. Specially the FOSS localisation not only lets users to use the software in local language, but also allows users to have the software free of charge [2]. Being a developing country Sri Lanka, is estimated to have an overall English Literacy of around 20% while the overall literacy rate is 90.6% [4]. Therefore the local language computing definitely reduces the digital divide that was caused due to the language barriers in Sri Lankan context as well. There are many efforts that have been made to localise Software in Sinhala and Tamil. In Sinhala there are lot of efforts to localise Operating Systems and Application Software. Presently FOSS application software such as Mozilla products, Joomla!, Moodle, GeoGebra, Squirrel Mail etc are being translated in to Tamil. Also localised versions of supporting materials are prepared in both languages [12].

Tamil localisation in Sri Lanka

Tamil is an official language in India, Sri Lanka and Singapore [10]. However Tamil people are scattered all around the world. There are many efforts that have been made to localise system software and application software in Tamil all around the world. Locales ‘ta’ and ‘ta_IN’ or ‘ta-IN’ were there when we entered to the localisation arena.

Glossaries are identified as a key element in localisation which helps to translate the strings. India – Tamil Nadu and Sri Lanka follow different IT – Tamil glossaries and therefore for a single IT term, we may get two different translations. There was a glossary published by Sri Lankan Official Language Commission with the collaboration of Indian scholars and Sri Lankan scholars. Even in that collaborative effort for many IT terms two Tamil translations, one for Indian -Tamil Nadu and one for Sri Lanka, are given.

Style guide is another important element in localisation and we had some disagreements with the style guides that were used in other Tamil localisation.

There are some features that were not very appropriate in the Sri Lankan context. For example, in Mozilla Firefox we can define custom features like search engines, feeds etc which are local to a country. Search engines that are defined for India will help to search Indian news and matters. Therefore there was a need to go for new locale.

In addition to that, some organisations strictly follow the combined version of ISO 639-1 and ISO 3166 to define locales. For example Joomla! is one of such organisations, which defines locale in this combined format.

Due to the uniqueness in glossary, style guide, technical requirement and standard policies the Tamil localisation efforts that are being taken in Sri Lanka are identified using the locale

name ta-LK or ta_LK. The ISO 639-1 language code alone is not meaningful enough to define the locale name.

Many application software are localised into Tamil language and it has been used in Sri Lanka as well as in other parts of the world. Most of the Tamil Localisations are being done at the University of Moratuwa, Sri Lanka. Not only the software GUI Localisations but also the user manuals are being prepared.

Localisation Process

At the University of Moratuwa we practice a particular localisation process similar to those who are mastering the field of localisation in other parts of the world. Here most of the phases of the process are handled using Pootle, a FOSS tool. The detail phases in the localisation process can be given as follows:

i. Identify the Software that needs to be localised and analyse the feasibility of localising it. Contact the respective organisations and inform them
ii. Get the language source files and identify the appropriate tools to do the translation
iii. Assign tasks to translators and get untranslated strings translated. Initially they use translation memory to translate the language strings.
iv. Review the translated strings, mainly for spelling mistakes and policy mismatch
v. Package the translated files and compile the localised version of the software
vi. Get the localised version of the software reviewed by people who are really going to use that software
vii. Submit the localised version to the respective organisation and make the localised version available to the public
viii. Prepare the required supporting materials like user manual or short guides
ix. Spread local applications and take them to end users
x. Maintain the language packs and update them with the new versions and feedbacks from end users

The software is identified depending on the requirements and most of the time FOSS applications are selected. Next the localisation methodology is analysed. If it is feasible to do the localisation then it is initiated. If there are many modules to be localised then only the essential parts of the modules are identified at the first phase of localisation. After that the tools are selected based on the format of the language source files. There are software for which we may need to use their own translation IDE. However, most of the language source files support for PO format. Therefore Pootle is being used to handle the localisation process in our context.

Once the language source files are identified then those files are added to the Pootle server and they are assigned to the translators. Then the translators download those files to their local machines and do the initial automatic translation using the translation memory. Then they carry on and translate the untranslated strings using our glossary and the style guide.

The IT- Tamil glossary published by Sri Lanka Official Language Commissions in 2000 is outdated now. Not only a lot of new terms have introduced after the year of 2000 but also some of the existing translated words in that glossary are not very appropriate. Therefore, based on that glossary we are continuously building a new set of terms and also we add new terms to the new glossary that we are building. For preparing this glossary we follow a separate process and that is not in the scope of this paper.

We also have our own style guide according to which our translators do the translations.

Compared to other style guides that are used in Tamil localisation our style guide has some notable differences. These are due to our contextual need. The main differences are,

- Provide access keys in English

  This is because Sri Lanka is a country where three languages including English are in use. Therefore producing a keyboard in one language is not feasible, especially for government and public sectors. The standardised keyboard is a trilingual keyboard. If we provide access keys in Tamil it may be difficult to switch to Tamil language before each time access the menu. This may increase the work for user rather than reducing it.

- Write English acronyms in English itself, but may write them in Tamil within braces if necessary

  This is because the transliterated form of English acronym may give funny meaning in Tamil. Therefore we write them fully in Tamil or let them in English.

- Do not transliterate and write names in Tamil, but may write transliterated version in braces.

  Again we do not transliterate names as they may give wrong pronunciation. But if it is really necessary we give the transliterated version in braces.

Once the translation phase is over the translated strings are reviewed. A team is formed for this and it reviews the translated strings. Then only it is built into the final product. Again the localised product is given to users who are really going to use that software. Then with the suggestions from the reviewers the translation is committed to the relevant organisations and released for public use.

Along with the translation of the software, the supporting materials such as user manuals or very short guides are also prepared in Tamil to provide support to a novel user. The prepared localised manuals are also released to the public.

A number of roles should be played by a team of people throughout this localisation process. Translation project managers, project owners, translators, linguistics, manual authors, supervisors and end users are the main roles in our localisation process. Moreover none of these works is one time work. Hence the process continues by maintaining the translations and taking feedbacks from the end users. These feedbacks are incorporated in the successive releases of the software.

However much effort is put into a localisation work it has less or no worth until it reaches the proper audience. This crucial phase is not practiced in any other localisation process. Apart from carrying out the localisation, we also do local application awareness programs all around the country. Several such programs have been held successfully in Schools and in Universities in Sri Lanka. The aim of these awareness programs is not only to introduce the localised version of the software but also to motivate users to use the local applications. In some situations we also give rewards to motivate users.

Success stories

There are many FOSS software that have been translated to Tamil language and they are being used by a range of users in Sri Lanka, from schools kids to university students. There are people who have come up with web sites after the introduction of local applications. Also as trainers we could see the enthusiasm of the people about localised software during the awareness sessions [12]. Mozilla Firefox, Mozilla Thunderbird, Moodle, Joomla!, SquirrelMail, Horde, GeoGebra are some applications that have been successfully translated using the localisation process that is discussed in this paper.

Not only these software but also the user manuals for these applications have been prepared. These are available in electronic format [12] and also in hard copy format. We continuously revise and update the manuals as well as the software..

Another success factor of our localisation efforts would be the number of hits that these software releases get over the internet. Among the above mentioned software Mozilla Firefox, Joomla! and Moodle have got the popularity all over world. These have been used by many people in Sri Lanka as well as people living in other countries. Out of the above mentioned software, Mozilla Firefox got the highest success and it was downloaded by more than 21 000 people [9].

Conclusion

Through the experiences and outcomes we can say that our localisation process is a successful one. Specially, taking the local applications to the people is a very important phase in it. This will increase the usage of local applications as well as improve the computer literacy.

Acknowledgement

We thank the LAKapps team members of University of Moratuwa, Sri Lanka for the work that they do on localisation and for their support. We also thank ICTA Sri Lanka and LK Domain for funding these efforts.

References

[1] Anousak Souphavanh, Theppitak Karoonboonyanan, 'Free/Open Source Software: Localization', Asia-Pacific Development Information Programme, 2005.

[2] S.W.Q. Jaffry,U.R. Kayani, 'FOSS Localization: A Solution for the ICT Dilemma of Developing Countries', 9th International IEEE MultiTopic Conference, 2005.

[3] What is localization?, http://www.lisa.org/Frequently-Asked-Que.46.0.html. Accessed : 2009-08-05

[4] Wasantha Deshapriya, 'Sri Lankan Country Report on Local Language Computing Policy', PAN Localization.

[5] Microsoft Language Portal, http://www.microsoft.com/language/en/us/download.mspx. Accessed : 2009-08-05

[6] Localization process, http://www.project-open.com/whitepapers/localization/l10n_biz_view.html. Accessed : 2009-08-05

[7] English country names and code elements, http://www.iso.org/iso/english_country_names_and_code_elements. Accessed : 2009-08-05

[8] Localization process, http://developer.apple.com/internationalization/localization/process.html. Accessed : 2009-08-05

[9] Mozilla addons, Tamil (LK) Language Pack 3.0, https://addons.mozilla.org/en-US/firefox/addon/6651. Accessed : 2009-08-05

[10] Tamil language, http://en.wikipedia.org/wiki/Tamil_language. Accessed : 2009-08-05

[11] Codes for the Representation of Names of Languages, http://www.loc.gov/standards/iso639-2/php/code_list.php. Accessed : 2009-08-05

[12] www.lakapps.lk. Accessed : 2009-08-05

[13] Open Office –Wiki, http://wiki.services.openoffice.org/wiki/Pootle_User_Guide, Accessed : 2009-08-05