Research Report on SeaMonkey Localization

Rafiq Ullah Kakar, Sharif Ullah Mahboob, Habib-ur-Rahman Ghaznawe
Afghan Computer Science Association (ACSA), Kabul, Afghanistan
rafiqullah.kakar@acsa.org.af , sharifullah.mahboob@acsa.org.af ,
habib.ghaznawe@acsa.org.af

Abstract

This document is a research report on SeaMonkey localization conducted by Afghan Computer Science Association. This document gives a brief picture of the SeaMonkey Localization process.

SeaMonkey is a free, open source and cross platform internet suit. It functions as web-browser, advanced e-mail and newsgroup client, IRC chat client, and HTML editor.

1. Introduction

After it was decided that localization of an application set that facilitated access to Internet content was required, SeaMonkey was selected because: 1) it was a complete Internet suite including a web browser and an e-mail client (because a browser and e-mail client was integrated into a single application, the overall localization effort would be less as compared to localizing a browser and e-mail client separately), 2) it had Unicode (UTF-8) and bidirectional language support which was required for Pashto, 3) it was localizable, and 4) it was supported on Windows, Linux and Mac OS X. SeaMonkey is an extension of the Mozilla Suite based on which the popular Firefox browser and Thunderbird e-mail client are being developed. Firefox and Thunderbird are standalone applications whereas SeaMonkey integrates several Internet applications. It was also ensured that SeaMonkey was a well-supported application based on activity in user and developer forums and mailing lists.

For Pashto, right-to-left text and bidirectional support had to be considered in addition to the encoding issue, because Pashto text (which uses Arabic script) is written from right to left but its numerals are written from left to right direction (making the script bidirectional). Also, strings to be translated may also include placeholders (e.g., ?%S?, where a string will be inserted by the application at run-time) which are also written from left-to-right.

2. SeaMonkey Localization Objective

Access to and generation of information through Information and Communication Technologies (ICTs) are yielding massive benefits, but since the medium of most of this information is English, it cannot be utilized by people who do not speak, read or write English. To bridge this digital divide and to truly harness the potential of ICTs a primary step is to localize available technology in indigenous languages. Localization of software involves adapting it for use by a particular culture. The core of any culture is its language so

language adaptation is a major part of the localization process. Localized SeaMonkey, is therefore a free, open source and cross platform internet suit that will serve for all Afghan's internet needs in their own native language interface in one application given the fact that it is a web-browser, advanced e-mail and newsgroup client, IRC chat client, and HTML editor.

In general, Localization of OSS can play a vital role in making ICTs available for indigenous populations because once the technology is available, it will support both the generation of content in local languages and access to it as well.

3. Localization Tools

The localization tools selected were Mozilla Translator and OmegaT. Both adequately provided the required language support for Pashto. Mozilla Translator was used for string management and translation incorporation and OmegaT was used for translation management.

4. Localization Registration

Currently the SeaMonkey registration process only requires making an addition to the Mozilla Wiki for SeaMonkey localization teams (http://wiki.mozilla.org/SeaMonkey:Localization_Teams), but as noted the process will be changing in the future and re-registration may be required when it does. Subscribing to the mailing lists for SeaMonkey development (dev-apps-seamonkey@lists.mozilla.org) and Mozilla localization is also recommended. Help can also be obtained at the IRC channels #seamonkey and #l10n at irc.mozilla.org

5. Translation Resources

The following resources are being used by the linguists during the translation process:

1. Core Glossary: This is a terminology glossary based on the Electronic Dictionary of Localization of Computer Applications (English - Pashto), by the Academy of Sciences Pashto Department for the localization of Microsoft Office and Microsoft Windows. This is being extended with the collaborative effort of both linguists and developers. This is the main glossary that the linguists refer to when translating as it represents the recommended standard for Afghanistan.

2. Urdu Core Glossary: This is a terminology glossary based on the Electronic Dictionary of Localization of Computer Applications (English - Urdu), by the National Language Authority Islamabad, Pakistan. This was provided to us by the Regional Secretariat Lahore.

6. String Extraction

The SeaMonkey GUI alone consists of over 10,000 strings. String extraction and preparation of batches could have proceeded in one of two ways, either proceed with each menu of the application (which is easily done as the translation files are organized that way) and translate 10,000 strings, or extract only unique strings (which can easily be done using Mozilla Translator and any spreadsheet application) which reduces the strings to a little over 6000. A disadvantage of the second way is that sometimes in the context of the application, an English word may possibly have different Pashto translations; but errors like this can be corrected at the testing stage, so it was decided to extract only the unique strings and translate them. A total of 55 batches for translation were created, where each batch contained about 600 words for translation.

7. Translation Review and Incorporation

Most translation errors detected during translations reviews were due to misinterpretation of the source string. This misinterpretation was caused by the following reasons; firstly due to limited exposure to software in general linguists were not familiar with some types of sentence structures used in software GUIs; secondly because the linguists had not used the software being localized, they could not understand concepts specific to the software (e.g., the notion of tabbed browsing), and might translate them inappropriately. In the case of SeaMonkey these errors can possibly be reduced by providing the linguists with SeaMonkey so that they can use it in place of the browsers and e-mail clients that they currently use.

8. Final Localized Product

After the localization is complete an announcement of the completion of the localization will be made and the localization will be provided and a request for it to be incorporated into the official project page for SeaMonkey will be made on the list. For SeaMonkey

the finalized localized product can be installed for Windows, Linux or Mac OS X, or a language pack known as an XPI which can be installed on a pre-installed SeaMonkey (on either Windows, Linux or Mac OS X). An XPI will be made since it can be used with all the official installers provided on the official SeaMonkey website. An Windows installer may also be provided, since our intended end-users will be using a Windows environment.

9. Conclusion

Possible improvements include integrating the use of a versioning system into the process, so that error correction is less tedious. Also, some sort of formalized peer review process between linguists (currently translations done by linguists are officially reviewed by the developers only) may be devised which may result in higher quality translation.

After the localizations are complete, it is recommended that the localization and possibly a glossary for that particular localization should be made available so that when a newer version of the application is released, the localization can be updated quickly and conveniently (even if the ownership for the localization changes). There should be no string changes for minor releases, but the localized product still needs to be updated. Each time there is a major release; additional strings for translation should be expected. The glossaries can also be used for reference in other localization projects.