Asian Journal of Information Technology 16 (7): 626-631, 2017
ISSN: 1682-3915

Arabic Localization of Open Source Software-A Case of Open Journal System

Samer Barakat
Department of Management Information Systems, Applied Science Private University,
Amman, Jordan

Abstract: Open source software systems gained popularity by many organizations in recent years. Open source systems vary from document management systems, content management systems to e-Learning systems. The low cost and ease of installation, use and customization of the systems has made it the right choice for many information systems departments as a reliable and low cost viable solution. In the research and the academic fields there are many open source software applications available on the World Wide Web for free download and use. The open journal system is one of the best and widely used systems developed by the public knowledge project. Functioning as a journal management and publishing system the system is developed as a nonprofit project and is an initiative of several universities working together for the development of free open source software and conducting research to improve the quality and reach of scholarly publishing CODATA. The open journal system support the use of several languages, however it does not have a ready translation file for Arabic language nor does it support the right to left text and document flow. Working on customizing the system into an Arabic version requires additional expertise in terms of customizing the cascading style sheets files and knowledge of the PHP programing language. The aim of this study is to provide a road map for developers to localize the open journal system to Arabic language using two main techniques. The localization effort has been centered on translating the local files. The customization of the cascading style sheet files and the layout templates shall be visited in another study due to the complexity of the process. The studys is intended to be a reference for all parties interested in the localization of the open journal system to be used by journals published in Arabic language.

Key words: Open source software, localization, open journal system, Arabic language, PHP programing language, non-profit project

INTRODUCTION

Open source systems are growing in acceptance by IT professionals in developing countries, since, they are easily used and available freely over the internet (Crosas, 2013). However, although, open source systems are easily localized into other languages, the process of Arabic localization includes additional measures that are not needed by other languages since, the Arabic script flows from right to left and this by itself requires working on the style sheets and the core files of the open source system.

Localization is the process of converting a software system into localized version holding the same functionalities and user interface however using a different language than it was originally built for. It is an adaptation process for the software system to be displayed in a different locale (King, 2007). The conversion process should be comprehensive to provide a meaningful interface for local users.

By localizing the open journal system into an Arabic locale, it shall be more accessible by Arab universities and research centers since it can be used in their native language. The localization effort includes translating the system online help, instructions and user interface.

WHY DO WE NEED SOFTWARE LOCALIZATION

Most open source software systems are written in English and this by itself represents a barrier for users from the Arab world. Additionally, open source software vendors are not interested in providing Arabic language support and locale, since, the market share for their products in the Arab world in very small compared to the users in Europe and the Western hemisphere who mostly speak English.

Therefore, the availability of an Arabic local for most open source systems depends on personal initiatives of Arab programmers that feel and see a need for such a system to be localized into Arabic and if no one from the Arab world took the lead in the localization effort it may be years before anyone would see a localized Arabic version of the open source software system.

The benefits of software localization can be harvested by all people in the Arab world. Providing a localized Arabic version of the open journal system shall help researchers, researchers, editors and publishing organizations to use the software without the need to learn English (Altman *et al*., 2001). They shall be using it in their own native language and this shall facilitate its use and the training process for new users.

Public and private universities, research centers and publishing houses can benefit from using the system without the need to have someone who speaks English to run the system and administer it. It shall provide Arabic researchers with a venue to publish their work in Arabic in a medium that is specifically designed for academic publishing, however it shall be in their native language (Altman, 2002; MacGregor *et al*., 2014).

OPEN JOURNAL SYSTEM LOCALIZATION

The open journal system was designed as a multilingual system, one that shall enable global publishers to publish their work in several languages. The public knowledge project the owner and the developer of the open journal system currently supports the following main languages, English, French, Spanish and Portuguese through their programming community (MacGregor *et al*., 2014). Other languages depend mainly on the efforts of volunteers interested in localizing the software to their native language.

Localization of the open journal system in Arabic requires working on certain locale files in XML format. These files are available in preset folders available on the installation directory of the open journal system. These language files are extracted from the system source code and stored in folders reflecting their International Standards Organization locale codes and country code. Therefore the main English language local files are stored in (en_US) folder. Other locales such as French and Spanish are stored in (fr_CA) and (es-ES) respectively.

The system was developed so that when it works on displaying a piece of text it needs to determine in what language it needs to be displayed (King, 2007). So, it goes to the system default language as set by the system administrator at the time of the installation and reads the default settings and based on that it starts reading from the corresponding local folder available in the system root folder.

The Arabic locale files has to be stored in an (ar_AR) locale folder. This folder shall contain the XML Arabic language files. Language files shall have lists of message keys tags. These tags include the massage key and its value in the template files of the cores system files. What is needed to create a localized version of the original local is to translate each line of the message keys in the locale files. For instance the following is a snippet of code form the local.xml file located in the (en_US) folder. The message keys and their explanation are displayed in English (Algorithm 1):

Algorithm 1:

```
<message key = "navigation.journalHelp">Journal Help</message>
<message key = "navigation.home">Home</message>
<message key = "navigation.about">About</message>
<message key = "navigation.userHome">User Home</ message>
<message key = "navigation.login">Log In</messag>
<message key = "navigation.register">Register</message>
```

To translate the file into Arabic we have changed all the wording in the message keys into the corresponding Arabic language translation of the English message as shown in Algorithm 2:

Algorithm 2:

```
<message key = "navigation.journalHelp">المساعدة</message>
<message key = "navigation.home">الرئيسية</message>
<message key = "navigation.about">عن النظام</message>
<message key = "navigation.userHome">صفحة المستخدم</message>
<message key = "navigation.login">تسجيل د خول</message>
<message key = "navigation.register">انشاء حساب جديد</message>
```

This processed needs to be done for all XML local files available for the system to use. The open journal system shall read these Arabic local files based on the default settings of the installed system and retrieve the message key value from the Arabic local files and display it on the screen. Whenever there is an error in the translation files and the system can't find the locale file or the message key in the local language it shall display the message key surrounded by hash marks. This is a mechanism used to indicate that the translation is not complete. ##navigation.home##.

INSTALLING THE ARABIC LANGUAGE PACK

The Arabic langue locale files have to be zipped in a tar.gz file archive format. Users interested in localizing their installation into Arabic must follow easy and simple steps. First thing they need to upload the Arabic langue locale files into their servers. Then they need to extract the content of the tar.gz file into the root directory on the server hosting the open journal system installation. This shall make sure that the Arabic locale files and folders are installed in the locale folder on the server. It is also needed to add the uploaded local to the system locales.xml file available in the registry folder on the server using the following tags in Algorithm 3:

Algorithm 3:

```
<?xml version = "1.0" encoding = "UTF-8" ?>
<!DOCTYPE locales SYSTEM"../dbscripts/xml/dtd/ locales.dtd">
<locales>
        <locale key = "de_DE" name = "Deutsch (Deutschland)"/>
        <locale key = "en_US" name = "English"/>
        <locale key = "ar_AR" name = "العربية"/>
        </locales>
```

Immediately after that the system administrator need to navigate to the language selection page on the open journal system and choose the check the box next to the Arabic local which is now registered and available on the language page. The next step is to click install to initiate the install of the new locale on the open journal system. The administrator can now choose the Arabic local from the list of supported locales and save and exit the page. Journal mangers can now activate the Arabic language from the language page and thereby converting the entire installation user interface into Arabic.

TRANSLATING THE OPEN JOURNAL SYSTEM

The open journal system can be translated using one of two methods. Either create a manual translation from scratch by opening each XML file and entering the Arab text in place of the English text or you can relay on the translation plugin already installed in the open journal system.

Creating a manual translation: For translating the open journal system manually you need to locate the local files in the installation directory of the open journal system root folder. There are several locations where the (en_US) folders exist. The first place to look under is the local/en_US folder on the root of the system. There are a number of XML files in this folder that needs to be translated into Arabic. In short there are five folders listed below where the local XML files that needs translation exists:

- The "locale/en_US" folder: in this folder you shall find several XML locale files. There is also the (email Templates.xml) locale file that needs to be translated into Arabic
- The "lib/pkp/locale/en"_US folder: in this folder you shall find several XML locale files with keys that is designed for all open journal system activities. We must translate all these files
- The "help/en"_US folder: this is where the help files for the open journal system are stored
- The "rt/en_US" folder: the reading tools for the open journal system are saved in this folder and they need to be translated as well
- And the "plugins/locale" folder: in this folder you need to translate the plugins that you are going to use in the system

Working on the files does not require using any software on your computer you can use the notepad text editor available on all windows machines or you may download and install and use the Notepad++ from the net and use it in editing your files.

Opening each file using any of the text editors reveals similar properties between each of them. You only need to work on the XML tags that define the language and the text showing up in the user interface. The main tag you need to change is the:

<locale name = "en_US" full_name = "US English">

This needs to be translated into Arabic as such:

<locale name = "ar_AR" full_name = "Arabic">

This needs to be done in all language and local files referencing the English language. After that you shall work on the real content that needs to be translated in the XML files. We shall explain how to translate the message keys available in the local.xml file as following Algorithm 4:

Algorithm 4:

```
<message key = "common.copy">Copy</message>
<message key = "common.preview">Preview</message>
<message key = "common.activate">Activate</message>
<message key = "common.deactivate">Deactivate</message>
<message key = "manager.review Forms">Review Forms</message>
```

We are going to translate into Arabic only the text that is between the tags. Therefore the following lines of code shall be translated as Algorithm 5:

Algorithm 5:

```
<message key = "common.copy">نسخ</message>
<message key = "common.preview">معاينة</message>
<message key = "common.activate">تفعيل</message>
<message key = "common.deactivate">ايقاف التفعيل</message>
<message key = "manager.review Forms">نماذج المراجعة</message>
```

This needs to be done for the entire XML file/s and then to be saved the new version in the local/ar-AR/local.xml. After translating all the necessary files, we have to make sure they are saved in the right location under the open journal system installation folder on the server. We also need to take into consideration that the right local folders are named according to the ISO locale codes for Arabic language. The new local must be

added to the registry local XML file in the (registry/local.xml) in order to allow the system administrator to install the Arabic local file properly.

Using the translation plugin tool: Newer versions of the open journal systems have been supplied with a translation plugin to allow system administrators and users to translate the system into other locales that are not supported by the current installation. Using the translation plugin made the translation process easier that before. The plugin is available for download if it was not already bundled with the original installation files. The plugin can be activated by loging-in to the journal manager and navigating to the user home and then choose system plugins:

- File browser
- Journal sections
- Review forms
- Languages
- Masthead
- Prepared emails
- Reading tools
- Setup
- Stats amd reports
- Payments
- System plugins
- Import/export data

The translator plugin is listed under the generic plugins. Once you navigate to the generic plugins page you click on enable link to make the plugin active. Afterwards you need to go back to the generic plugins page, scroll down to the plugin and click the newly-available translate link.

- Matadata plugins
- Authorization plugins
- Block plugins
- Citation format plugins
- Citation database connector plugins
- Citation output plugins
- Citation extraction plugins
- Gateway plugins
- Generic plugins
- Implicit authentication plugins
- Import/export plugins
- OAI metadata format plugins
- Payment plugins
- Public identifier plugins
- Report plugins
- Theme plugins
- Install a new plugin

Translator plugin: This plugin allows web-based maintenance of translation files enable upgrade plugin delete plugin.

Thesis abstracts: Solicit and publish thesis abstracts enable upgrade plugin delete plugin:

- Translator plugin this plugin allows web-based maintenance translate disable upgrade plugin delete plugin
- Web feed plugin

The translator plugin page shall open up showing all available locals. The English language local is at the top of the list in Fig. 1. Click Edit under action menu to enter the edit mode which lists the local XML files. You can edit each file by clicking the Edit link next to each file name (Fig. 2). The edit mode shows the key-value and action to be performed on each key (Fig. 3).

Now we can start the translation process by changing the text in the translation field form English to Arabic. The original text shall always be available under "Reference Version" so that the translation can always be enhanced if the case may be required. Figure 3 shows how the translation in the translation plugin is much easier that opening up the XML file and changing the text between the tags since, there are no tags to worry about deleting accidentally while performing the translation process.

The translated text shows up in the system using the Arabic language instead of English. The sentence "Call us" has been changed to its Arabic equivalent.

- Pople
- اتصل بنا
- Editorial team

Exporting translation: Translated files can be exported through the translator plugin as a compressed language folder. The folder can be downloaded as a package file to be used on other systems or in case you need to migrate your open journal system installation to a new server.

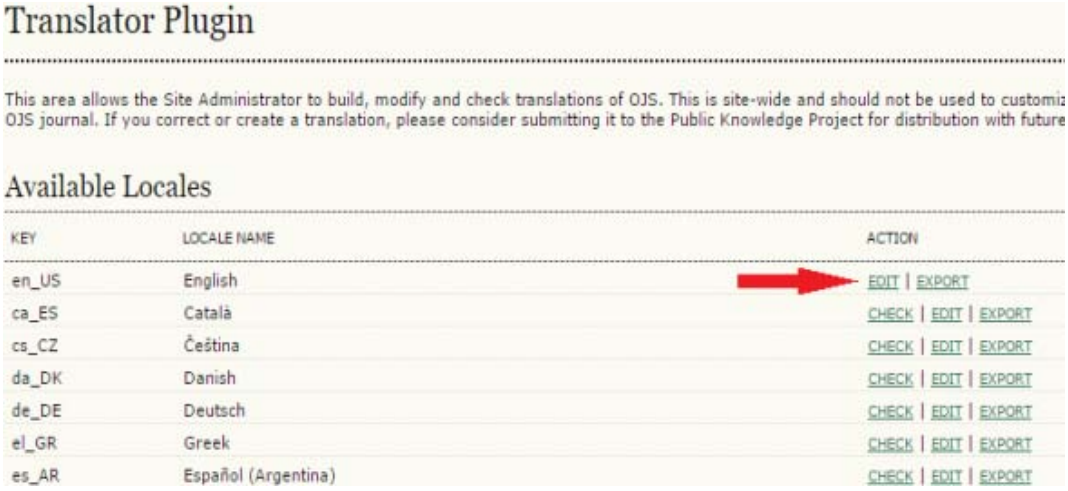

Fig. 1: Translator plugin

Fig. 2: Local files

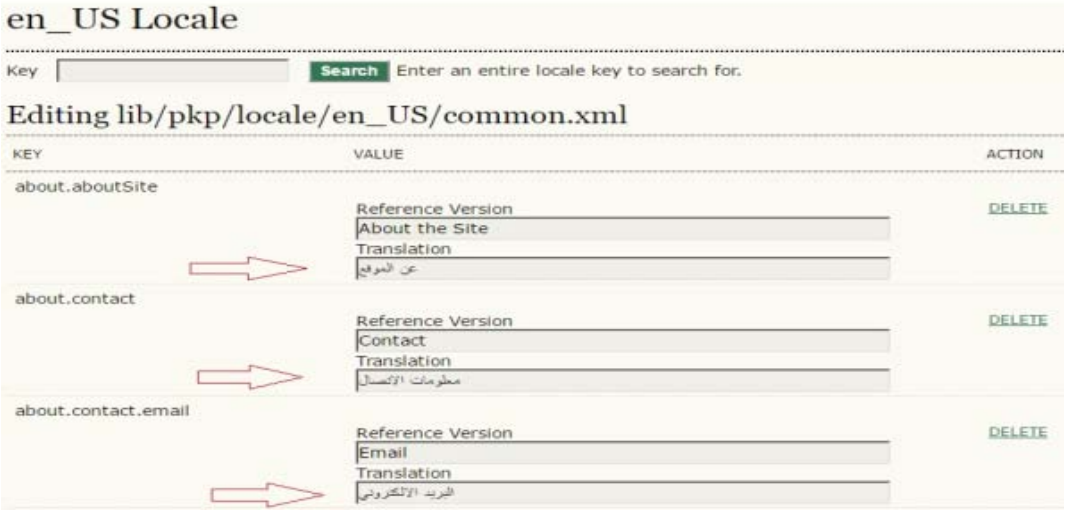

Fig. 3: Translation plugin

CONCLUSION

In this study we have provided a road map for the translation of the open journal system to Arabic language. The process was straight forward and fairly easy use by non-experienced programmers and system administrators. The study provided two main techniques for the translation from English to Arabic. The first one was using the manual translation approach and the second approach was using the translation plugin tool. Additional work on customizing the cascading style sheets files and the system PHP files was beyond the scope of this study. The customization of the cascading style sheet files and the layout templates shall be visited in another study due to the complexity of the process. The studys is intended to be a reference for all parties interested in the localization of the open journal system to be used by Journals published in Arabic language.

REFERENCES

Altman, M., 2002. Open source software for Libraries: From Greenstone to the Virtual Data Center and beyond. Iassist Q., 25: 5-12.

Altman, M., L. Andreev, M. Diggory, G. King and E. Kolster *et al.*, 2001. Overview of the virtual data center project and software. Proceedings of the 1st ACM CS Joint Conference on Digital Libraries, June 24-28, 2001, ACM, Roanoke, Virginia, ISBN:1-58113-345-6, pp: 203-204.

Crosas, M., 2013. A data sharing story. J. E-Sci. Librarianship, 1: 173-179.

King, G., 2007. An introduction to the dataverse network as an infrastructure for data sharing. Sociological Methods Res., 36: 173-199.

MacGregor, J., K. Stranack and J. Willinsky, 2014. The Public Knowledge Project: Open Source Tools for Open Access to Scholarly Communication. In: Opening Science, Sonke, B. and F. Sascha (Eds.). Springer, Berlin, Germany, ISBN:978-3-319-00025-1, pp: 165-175.