Internationalization in WebDynpro ABAP Applications

Applies to:

SAP ECC 6.0. For more information, visit the Web Dynpro ABAP homepage.

Summary

The article describes the concept and procedure of developing a Multilanguage capable Web Dynpro ABAP application

Author: Bijal Parmar

Company: Larsen & Toubro Infotech Limited

Created on: 8 September 2009

Author Bio

Bijal Parmar is a Software Consultant in Larsen & Toubro Infotech Limited. She has three years of SAP experience. She has worked extensively in ABAP, APO, Workflow and Visual Composer.

© 2009 SAP AG

Table of Contents

© 2009 SAP AG

Concepts

Internationalization - Purpose

The goal of Internationalization is to provide the technical foundation to enable programs to support multiple scripts and languages without redesign or modification. Once a user has selected a log-in language, all programs must transparently alter their run-time behavior to meet the expectations of the user. Internationalization furthers generic programming, because hard-encoded search strings, error messages, etc. may work in one language, but not in another.

- The purpose of internationalization is to adapt existing applications so that they can be used in several languages and countries (or regions) without the need for changes to the source code.
- Using the internationalization tools, you can store these text strings separately from the application source code, in a format that can be processed in the standard translation system.
- There should be no Hard Coding of text in the code
- Translations should be maintained for the desired languages.

Internationalization in WebDynpro ABAP

To enable the WebDynpro ABAP application to be displayed in the user log on language we need to apply the concept of internalization.

- Without Internationalization:

Web Dynpro ABAP Component is created in logon language. When the application is launched it gets displayed in the same language.

For instance, the WebDynpro ABAP component is created in English. When user login language is English, he/she will see the entire application in English. If the user login language is German, even then he/she will see the entire application in English.

- With Internationalization:

To let the user see the application based on his/her logon language, language translation should be maintained for UI elements and data.

For instance, the WebDynpro ABAP component is created in English. Also, the translations are maintained for GERMAN language.

When user login language is English, he/she will see the entire application in English.

If the user login language is German, he/she will see the entire application in German.

If the user login language is Chinese, he/she will see the entire application in English, since no translations are maintained for it and in such case the application gets displayed in Default language.

© 2009 SAP AG

Translation Elements

There are two things for which the translations will be maintained.

1. UI elements like Labels, Text Fields, Titles, Table Column headers.
2. Actual Data
   Example: Remote Function modules are called in the WebDynpro components using service calls. The data obtained for it must be made language enabled too. For this the RFC must fetch data based used logon language and all text must be in the same language.
   Example: Material Description obtained from R/3 database tables. Description will be marinated in required languages. When you select data from it, add the language parameter in where clause.

Online Text Repository (OTR)

- WebDynpro ABAP applications can be made language independent using **OTR objects**
- The Online Text Repository (OTR) is a central storage location for texts. The main task of the OTR is to support the creation and translation of texts

Developing a Multi-language Enabled WebDynpro ABAP Application

- Go to Transaction SE80 - Create a WebDynpro Component “ZMULTILANGUAGE”
- Create a View “MAIN”
- Create a UI element type Text View – TEXTVIEW1.
  - ID – TEXTVIEW1
- Create a OTR object for the UI element

  Click on GOTO-> Online TEXT Repository Browser

  Click in  a POP up will appear

  Enter details as below

  Save it.
- All OTR objects will be grouped based on the package in which it is saved. Here we have saved it as local objects. Hence the objects under $TMP in screen shots below are created for all UI elements.

  Also, each OTR object will have an alias name which is

  <package name>/<OTR_Object_name>.
- Bind the OTR object to the UI element in text property

Click on  . A POP Up as below is displayed with list of OTR objects

Select the appropriate alias we had created. This will bind the OTR object to the UI element

Note the text property is now bonded to the OTR object.

SAVE it.

- Create WebDynpro application

Give application name and description

Save and Activate the WebDynpro Component

- Maintain Translations
  - GO to SOTR_EDIT Transaction
  - Enter ALIAS name and click on Display
  - Since we created the OTR object with English as login language. In below screen use the same language

© 2009 SAP AG

In next screen, select Edit-> Context->Change

Click on  , the fields will now be editable.

Select GERMAN Language. Fill the text field with German text for “Hello World!”. Save it.

In text field,

Hello World! – In English

Hallo Welt! – In German

Save it.

Thus, we have maintained text for two languages – English and German. The Alias name is same for both.

Output

- Login language is English

Go to SE80 – Test the Application

Application opens in IE

- Login language is German

Go to SE80 – Test the Application

Application opens in IE

Based on the login language of user, the text is translated using OTR objects.

Summary

Thus, we have understood the need for Internationalization and the method to implement it for WebDynpro ABAP applications.

© 2009 SAP AG

Related Content

https://wiki.sdn.sap.com/wiki/display/stage/OTR+Text+Translation+Tool+for+Web+Dynpro+ABAP

https://wiki.sdn.sap.com/wiki/display/WDABAP/Translating+Texts+with+SOTR

https://forums.sdn.sap.com/thread.jspa?threadID=1241288

For more information, visit the Web Dynpro ABAP homepage.

© 2009 SAP AG

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.

© 2009 SAP AG