

ADDING THE KURDISH LANGUAGE TO VB .NET GLOBALIZATION: MAKING A CASE AND TECHNICAL CODE

Azad Ali, Indiana University of Pennsylvania, Azad.Ali@IUP.edu
Seever Sulaiman, interthinks, ssulaiman@interthinx.com

ABSTRACT

This paper is about adding Kurdish to the list of languages supported in Visual Basic .NET (VB .NET) globalization. The introduction of Visual Studio .NET (VS .NET) has simplified translating programs from one language (or culture) into other languages (or cultures). This translation is accomplished through a set of procedures and data available for a list of languages supported in VS .NET. However, the list of languages that VS .NET supports does not include the Kurdish language. This paper is to argue for adding the Kurdish language to the list of languages supported in VB .NET globalization. The paper then utilizes a new class that made available in Visual Studio .NET version 2005 to include some of the data for the Kurdish language.

Keywords: VB Globalization, VB New Languages, VB Kurdish, VS Kurdish

INTRODUCTION

The introduction of globalization in Visual Basic .NET (VB .NET) has significantly simplified the steps it takes to write a program for multiple languages. Through the use of classes, routines and data; application developers can write a program in one language (say English), then with minimum efforts, the same developers can translate the program into another language (say Arabic). This is mostly accomplished through a set of data and routines available for a list of languages supported in VB .NET globalization [1, 4, 5, 8]. However, the list of languages supported in globalization is incomplete and is missing many languages. Among the languages that are missing from VB .NET globalization is the Kurdish language; thus it is depriving Kurdish users from using VB .NET to display applications in their own language.

This paper is to argue and make a case for adding the Kurdish language to the list of languages supported in VB .NET. The paper lists also some technical code for adding Kurdish as a customized language. Starting with Visual Studio .NET version 2005, VB .NET programmers can use newly created classes to add their own customized language to the list of

globalization languages supported on their computers. Thus the authors include a program written in VB .NET to accomplish this task of adding the customized Kurdish language. A summary and suggested future work is included at the end.

ABOUT GLOBALIZATION

Globalization in VB .NET in its simplest term means the steps and processes it takes to translate a program written in one language (say, English) to another language (say, Arabic). Prior efforts to translation used to face difficulties due to different storage methods and due to disparity in representation and processing of data in different languages. Programming writers identified five areas that contributed to this difficulty of translating programs among languages [1, 4, 5, 8]:

- 1- The display of text information: Different languages may use different alphabets and this may change the display of text in programs.
- 2- Date and time format: various languages utilize different calendars and this affects the number of days per year and the display and calculations related to dates.
- 3- Use of currency symbols is different depending on the country being used.
- 4- The sort order is different from one language to another depending on the character representations in different languages.
- 5- Display of specific data like postal code and the different formats used.

VB .NET simplified writing programs for multiple languages by providing data, routines and classes for a list of languages that it supports in globalization in order to resolve some of the problematic areas that are listed above [1, 4, 5, 8].

VB .NET globalization works by providing formatting for countries, languages and cultures in the processing of globalization data. Since there could be more than one language spoken in a particular country, VB .NET distinguishes between cultures and languages for the representation of their data. A language like Arabic is spoken in different cultures like Iraq, Syria and Saudi Arabia. Ekedahl

[4] uses an example in VB .NET to list the data that are stored for various languages and cultures. The display of the output from this program is shown in Figure 1. As the user clicks one of the cultures, the program displays the data specific to that language/culture.

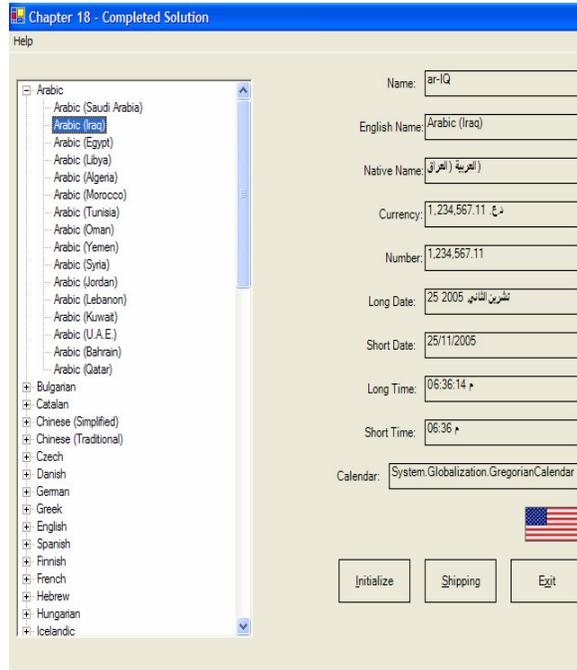


Figure 1. Data Display for Cultures Using a Program in VB .NET Ekedahl 2004 [4]

Figure 1 displays data about date, time, number format and others in the Arabic language for the Iraqi culture after the user clicked on the language and culture combination. VB .NET follows a standard code for identifying culture/language similar to standard of languages followed by International Standardization Organization (ISO). The ISO is an organization responsible for developing international standards. Among the standards developed by the ISO are the different codes for identifying language and culture uniquely. Each combination of culture and code is given a unique code that it is distinguished from other cultures. So the example in Figure 1 shows the data for the Arabic language and Iraqi culture that is given the code ar-IQ.

MAKING THE CASE FOR ADDING THE KURDISH LANGUAGE

The Kurdish language is spoken by millions of people who live in an area known as Kurdistan. The boundaries of Kurdistan stretch to include portions of Iraq, Iran, Syria, Turkey, Azerbaijan and others [9]. The Kurds are distinguished from other surrounding

cultures by their own language, culture, geographic locations and other characteristics. Yet, despite the numerous differences between the Kurds and other cultures, their representation in international settings has been underrepresented and the Kurds are often grouped with other cultures in different international settings. One of the settings in which the Kurds are underrepresented is in VB .NET supported languages.

The Kurdish language is not among the languages that are supported in VB .NET. So Kurdish users of VB .NET cannot by default display programs in their language; instead they have to display the program in other languages like Arabic, Turkish and Farsi. Microsoft grouped the Kurdish language with the languages that are supported with the Arabic alphabet [3]; thus the implication is that the Kurdish users display their programs in the Arabic language. This section is to argue for adding the Kurdish language to the list supported in VB .NET globalizations. It will base its argument on factors of the ISO code, alphabet, population, the notion of country/nation and other factors.

The ISO Code

The International Standards Organization has developed two sets of standards for coding languages and cultures: ISO 639.1 and ISO 639.2 [6]. The list of languages supported in both codes exceeds 500 compared to 126 languages supported in VB .NET [3]. The Kurdish language is listed under both versions of ISO code and is given the codes ku and kur, respectively. Yet VB .NET did not include the Kurdish language and instead included languages that are not listed in ISO 639 or languages that are listed only in ISO 639.1. For example the language Syriac in Syria and the Konkani both are not listed in ISO639.1 but they are supported in VB .NET. At the same time, VB .NET supports some languages like the Basque and the Kyrgyz (in Kyrgyzstan), yet these languages are not listed in ISO.

The Kurdish Alphabet

Kurdish is grouped with the Arabic language in VB .NET globalization setting because both languages use the Arabic alphabet [3]. At the same setting, the Farsi (Persian) language is supported separately from the Arabic even though it (Farsi) uses the Arabic alphabet. Additionally there are still differences between the Kurdish and Arabic languages regarding the dates and the number and sort order of characters in the alphabet. The month names are different in Kurdish from Arabic and the Kurdish alphabet

contains 36 letters [11] compared to 28 in Arabic [10]. Then the sort order is handled differently in VB .NET for combined characters in the alphabet [5]. Kurdish uses combined characters for some of its alphabet when it joins some two characters into one, while Arabic does not include such combined characters [11].

The Population of the Kurds

The population of the Kurds varies depending on the studies conducted, and it ranges from 26 million [11] to 39 million [9]. The population of the Kurds in Iraq exceeds 5 million [11]. This number of people exceeds the population of people in many languages/cultures that are supported in VB .NET. The World Factbook by the CIA lists the population of 239 countries in descending order [2]. The higher estimate of the Kurds puts them at the 34th rank while the lower estimate puts them in the 47th rank among the populace countries listed in the CIA Factbook. Even if only the Kurds in Iraq are counted, they are still ranked 105 on the population list. Yet, there are many countries supported in VB .NET but fall below the Kurds in the rank of the populations list. Among the countries that are supported in VB .NET but are less populace than the Kurds include Kyrgyzstan (5 million), Georgia, Croatia and Lebanon (4 million each), Albania (3 million), Kuwait (2 million), Qatar (1 million), Iceland (quarter of a million) and others [2].

The Notion of County/Nation

Listing the country name in VB .NET may give the notion that the list of languages it supports may be dependent on countries or nations by some kind of definition of these terms. But this notion is not followed in VB .NET either. The VB .NET number of supported language is 126, yet the UN lists 191 nation members [7] and the CIA lists 234 countries that include members and non-members of the UN [2]. But any one of these numbers indicates that the VB .NET list is not based on the notion of country/nation. It is a given fact that the Kurds do not have an independent nation, but this still is not a justification for excluding the Kurdish language. An

added note here is that the VB .NET globalization list includes the Arabic language in 15 different countries while the number of Arab countries is 21, thus missing some countries like Sudan, Somalia, and others.

The Notion of Minority versus Majority

Kurdish exclusion from VB .NET may be dependent on the notion of majority versus minority in countries. Since the Kurds are not a majority in any of the countries in which they live, they may be excluded from VB .NET supported languages accordingly. But this notion is incorrect. For example, VB .NET supports the English culture in the Philippines, and it is not a language spoken by the majority. Also, VB .NET supports eight languages spoken in India (Punjabi, Gujarati, Tamil, Telugu, Kannada, Marathi, Sanskrit, and Konkani) and not all of them can be a majority. Yet, VB .NET does not support the Kurdish language even though it is largest ethnic population in at least four countries [9].

ADDING THE KURDISH LANGUAGE USING VB .NET 2005

The test of the program in this paper has been conducted on VB .NET 2003. According to what we have found, new customized cultures cannot be added to VB .NET 2003 or earlier versions. However, we learned that VB .NET 2005 introduced a new class that enables programmers to add their own customized culture. At the time of writing this paper, VB .NET 2005 was not introduced into the market, but we were able to download a beta version temporarily. We used a new class in VB .NET 2005 for adding customized cultures and were able to add the new customized Kurdish culture. Figure 2 below shows the execution of the same program used in Figure 1, but this time it shows the Kurdish culture displayed among the cultures. The Kurdish culture was included after executing a program that we created in which we were able to add the new customized culture. Appendix A at the end of this paper lists the code that we used to add the Kurdish culture.

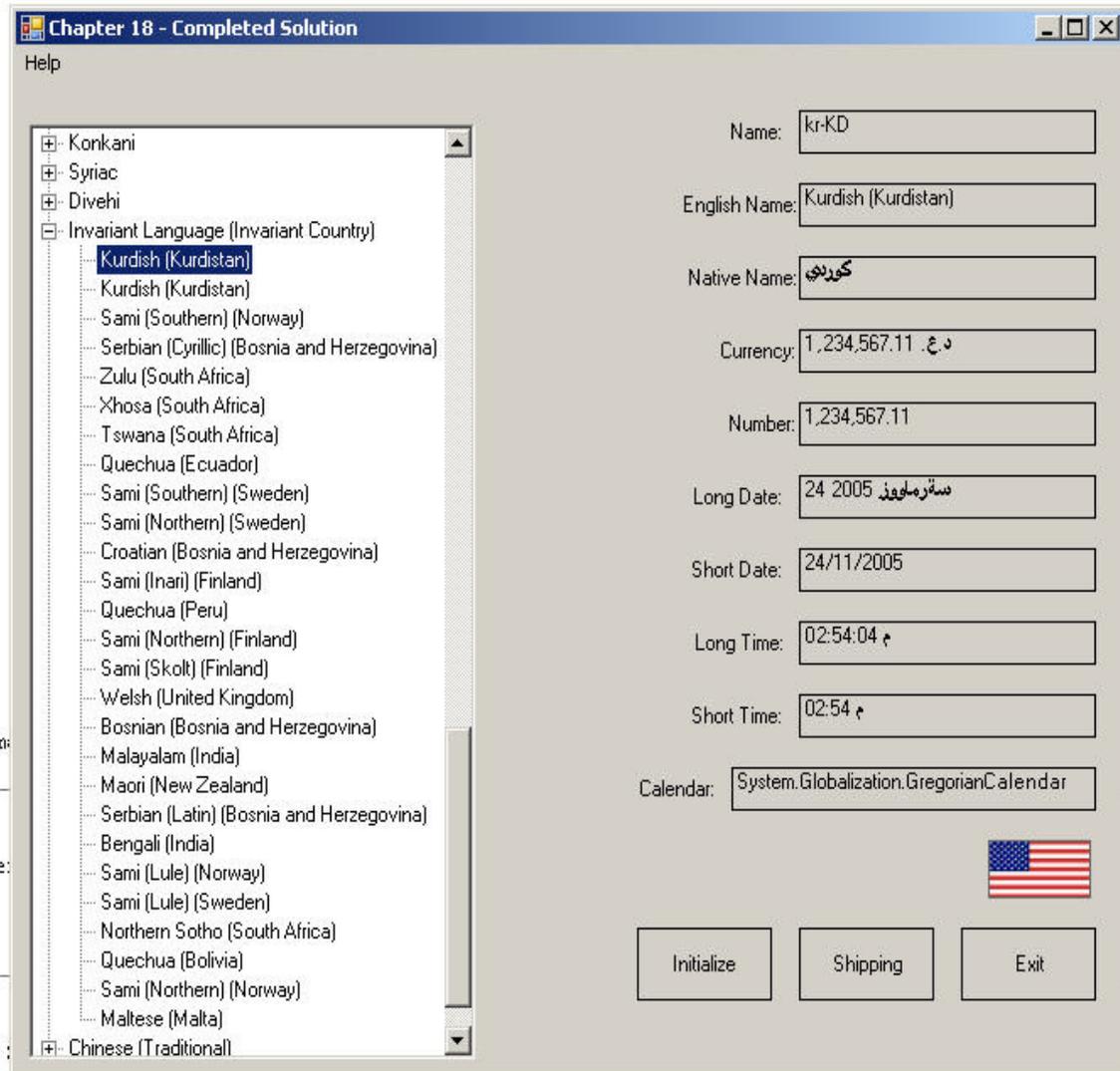


Figure 2. Display of Program Execution after Adding the Kurdish Language

SUMMARY AND FUTURE WORK

This paper described the addition of a new language to the VB .NET globalization setting. It attempted to make the case for adding the Kurdish language based on features of ISO, alphabet, population, notion of country/nation and other factors. The paper also listed a solution that works around the limitation of default VB .NET setting and was able to add the Kurdish language. This solution is based on a class introduced in VB .NET 2005.

The page limitation imposed on papers for this conference restricted us from making our case clearly and also limited us from explaining in full our solution to work around and introduce the Kurdish language. Thus the authors here are planning to write

another paper on the same subject. The new paper is going to make the case more clearly for the Kurdish language. The suggested paper will also explain more fully the technical aspects for the solution to adding a new customized language/culture to VB .NET globalization.

REFERENCES

1. Bradley, J; Millspaugh. A (2003). *Advanced Programming Using Visual Basic .NET*. Boston: Irwin McGraw Hill.
2. CIA – The World Factbook (2005). *The World Factbook: Rank Order – Population*. Retrieved November 28, 2005 from <http://www.cia.gov/pulications/factbook/rankorder/2119rank.html>

3. Dr. International (2005). *Ask Dr. International*. Retrieved November 28, 2005 from <http://www.microsoft.com/globaldev/DtIntl/columns>.
4. Ekedahl, M (2004). *Programming Guide to Developing and Implementing Windows-Based Applications with Microsoft Visual Basic .NET*. Reno, NV: Thomson Course Technology.
5. Gunderloy, M (2003). *MCAD/MCSD Developing and Implementing Windows-Based Applications with Visual Basic .NET and Visual Studio .NET*. Indianapolis: QUE Publishing
6. ISO 639.2 (2005). *Codes for Representation of Names of Languages*. Library of Congress. Retrieved November 15, 2005 from <http://www.loc.gov/standards/iso639-2/englangn.html>
7. List of Member States (2005). *List of Member States*. Retrieved November 28, 2005 from <http://www.un.org/overview/unmember.html>
8. Kaplan, M (2000). *Internationalization with Visual Basic. The Authoritative Solution*. Indianapolis, IN: Sams Publishing.
9. Kurdish Population (2005). *The Distribution of the Kurdish Population*. Retrieved November 28, 2005 from <http://www.geocities.com/kurdishcommunityofotawa/population.htm>.
10. Omniglot [1](2005). *Arabic alphabet, pronunciation and language*. Retrieved November 28, 2005 from <http://www.omniglot.com/writing/kurdish.htm>.
11. Omniglot [2](2005). *Kurdish Language, Alphabet and Pronunciation*. Retrieved November 9, 2005 from <http://www.omniglot.com/writing/kurdish.htm>.

(Appendix on Next Page)

Appendix A. Programming Code for Adding the Kurdish Language

```
Imports System.Globalization
Public Class frmCultureBuilder
    Private Sub cmdCreateCultureFromScratch_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles cmdCreateCultureFromScratch.Click
        CultureAndRegionInfoBuilder.Unregister("ku-KU")
        Dim cib As CultureAndRegionInfoBuilder = Nothing
        Console.WriteLine("Create the CultureAndRegionInfoBuilder..." & vbCrLf)
        cib = New CultureAndRegionInfoBuilder("ku-KU", CultureAndRegionModifiers.None)
        cib.CultureEnglishName = "Kurdish (Kurdistan)"
        cib.RegionEnglishName = "Kurdistan"
        cib.CultureNativeName = "ᠪᠠᠨᠢᠶᠢ"
        Dim cia As CultureInfo
        cia = New CultureInfo("ar-IQ")
        cib.NumberFormat = cia.NumberFormat
        cib.GregorianCalendarName = cia.CalendarName
        cib.TextInfo = cia.TextInfo
        cib.CompareInfo = cia.CompareInfo
        cib.KeyboardLayoutId = cia.KeyboardLayoutId
        cib.RegionNativeName = "ᠪᠠᠨᠢᠶᠢᠨᠠᠵᠤ"
        cib.ThreeLetterISOLanguageName = "kur"
        cib.ThreeLetterISORegionName = "kur"
        cib.TwoLetterISORegionName = "ku"
        cib.ThreeLetterWindowsRegionName = "kur"
        cib.ISOCurrencySymbol = "USD"
        cib.CurrencyNativeName = "Dinary Kurdi"
        cib.CurrencyEnglishName = "Kurdish Dinar"
        Dim cinull As CultureInfo
        cinull = New CultureInfo("")
        cib.Parent = cinull
        cib.ThreeLetterWindowsLanguageName = "kur"
        cib.TwoLetterISOLanguageName = "ku"
        cib.IetfLanguageTag = "kurdisk"
        cib.Register()
        cib.Save("C:\temp\kurdish.xml")
        Console.WriteLine("Create and explore the custom culture..." & vbCrLf)
        Dim ci As CultureInfo
        ci = New CultureInfo("ku-KU")
        Console.WriteLine("Name: . . . . . {0}", ci.Name)
        Console.WriteLine("EnglishName: . . . . . {0}", ci.EnglishName)
        Console.WriteLine("NativeName: . . . . . {0}", ci.NativeName)
        Console.WriteLine("TwoLetterISOLanguageName: . . . . . {0}",
ci.TwoLetterISOLanguageName)
        Console.WriteLine("ThreeLetterISOLanguageName: . . . . . {0}",
ci.ThreeLetterISOLanguageName)
        Console.WriteLine("ThreeLetterWindowsLanguageName: . . . . . {0}",
ci.ThreeLetterWindowsLanguageName)
        ci = Nothing
    End Sub
    Private Sub cmdCreateCultureFromXML_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cmdCreateCultureFromXML.Click
        Dim cib As CultureAndRegionInfoBuilder = Nothing
        CultureAndRegionInfoBuilder.Unregister("ku-KU")
        cib = CultureAndRegionInfoBuilder.CreateFromLdml("C:\temp\kurdish.xml")
        cib.Register()
    End Sub
End Class
```