**Multilanguage in WPF**

Einleitung:

Wenn die Anwendung ausschließlich in deutscher Spracher erhältlich , so beschränkt sich der Markt acuh haupsächtlich auf die Deutsch gesprochene Länder. Es wäre sehr schön, wenn die Anwendung in Englisch übersetzt wird. Und daher kann man wesentlich mehr potenzielle Kunden welweit erreichen. Es gibt aber eine Vielzahl an Benutzern, die Anwendungen sogar in ihrer eigenen Mutterspracher bevorzugen. z.B : Französisch,Spanisch,….

Die WPF stellt hierzu Techniken zur Verfügung, un Anwendungen an unterschiedliche Sprache und Kulturen anzupassen. Und hier wird die Technik „Lokaliersung“ vorgestellt.

Oberflächen Lokalisieren:

Um die Anwendung zu lokaliesieren, wird es Schritt für Schritt wie folgt ausgeführt:

## Schritt 1: Standardsprache für das Projekt setzen

In der Projektdatei „\*.csproj“ muss der Knoten UICulture unterhalb des Knoten ProjectGroup hinzugefügt werden. Mit Rechte-Klick auf den Projektdatei „\*.csproj“ und kann man den Datei mit beliebigen Editor geöffnet. Hier wird mit Editor Notpad++ geöffnet und Deutsch ist Standarsprache als Beispiel. Die Abkürzung „de-De“ steht für Deutsch.

<?xml version=“1.0“ encoding=“utf-8“?>

<Project>

<ProjectGroup>

**<UICulture>de-DE</UICulture>**

</ProjectGroup>

</Project>

Nachdem Kompilieren wird die Datei „\*.resources.dll“ (z.B: …. **UnisensViewer.resources.dll**) in der Ordner ….\Debug \de-DE erstellt.

## Schritt 2: Attribut setzen

Öffnen Sie die Datei „assembly.cs“ und fügen diese Zeile ein.

**[assembly: NeutralResourcesLanguage (“de-DE“,**

**UltimateResourceFallbackLocation.Satellite)]**

## Schritt 3: Oberfläche vorbereiten

Hierfür wird das Attribute namens „Uid“ (Unit identify) benötigt, das sich in dem XAML-Namensraum befindet. Entweder kann man den Attribute „Uid“ selber ein,

<MenuItem ***x:Uid="menu\_File"*** Header="\_Datei">

<MenuItem ***x:Uid="menu\_File\_New"*** Header="Neu" ... ... .../>

<MenuItem ***x:Uid="menu\_File\_Open"*** Header="Öffnen..." .../>

<MenuItem ***x:Uid="menu\_File\_Save"*** Header="Speichern" .../>

<MenuItem ***x:Uid="menu\_File\_SaveAs"*** Header="Speichern unter..."/>

<MenuItem ***x:Uid="menu\_File\_Close"*** Header="Schließen".../>

</MenuItem>

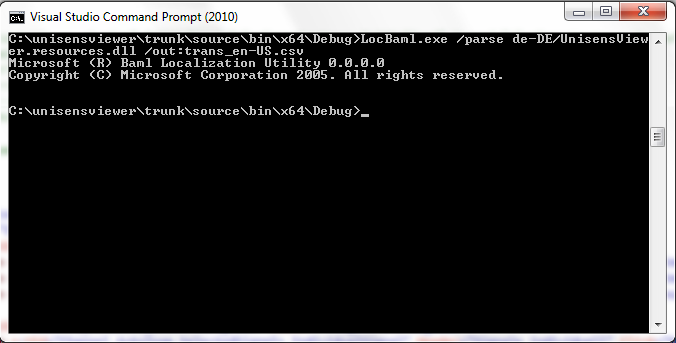
oder UIDs wird automatisch durch den Befehl „ msbuild /t:updateuid „ vergeben.

msbuild **/t:updateuid** UnisensViewer.csproj

## Schritt 4: Inhalte übersetzen

Das Programm LocBaml lässt sich über die Kommandozeile starten.In dem folgenden Beispiel wurde als Sprache Deutsch (de-DE) eingetragen, sodass die Resourcendatei in dem Unterverzeichnis „de-DE“ im Debug-Verzeichnis zu finden ist.

LocBaml.exe /parse de-DE/UnisensViewer.resources.dll /out:trans\_en-US.csv



Die CSV-Datei „trans\_en-US.csv“ kann nun mit einem bevorzugten Programm geöffnet werden und sie enhält 7 Spalte:

* BAML-Stream-Name
* Resourcenschlüssel
* Kategorie
* Lesbarkeit
* Änderbarkeit
* Kommentar
* Inhalt

## Schritt 5: Sprachabhängige Ressourcen erzeugen

Nachdem Übersetzung müssen aus den übersetzten wieder eine neue Resource mithilfe des Programm „LocBaml“ Text für das WPF-Projekt generiert werden. Der Aufruf von LocBaml soll wie folgt aussehen:

LocBaml.exe /generate de-DE\UnisensViewer.resources.dll /trans:trans\_en-US.csv /out:en-US /cul:en-US

Die neue Resource wird in der Unterverzeichnis …\Debug\en-US abgepeichert.

## Schritt 6: Sprache auswählen

Um die aktuelle Sprache beim Start des Programms zu verändern, werden die folgende Zeile in die Programm eingefügt.

Thread.CurrentThread.CurrentCulture = new CultureInfo(“en-US“); Thread.CurrentThread.CurrentUICulture = new CultureInfo(“en-US“);

* CurrentCulture: hier wird die Kultur “en-US“ für den aktuellen Thread festgelegt.
* CurrentUICulture: legt die aktuelle Kulture “en-US“ fest, mit deren Hilfe der Resourcen-Manager zur Laufzeit kulturabhängige Resourcen sucht.

Überzetzung der weitere Elemente ( Menüpunkte, Texte,…)

Wiederholen Sie die **Schritt 3, 4, und 5** um **weitere UID** hinzufügen und die **weitere Elemente zu übersetzen** bzw. **neue Resources-Datei** zu erzeugen.

Neue Sprache hinzufügen

Um neue Sprache( z.B. Vietnamesisch) hinzufügen, werden **die Schritt 4, 5, 6** wiederholt. Als Beispiel können wir wie folgt sehen:

**Schritt 4:**

**LocBaml.exe /parse de-DE/UnisensViewer.resources.dll /out:trans\_vi-VN.csv**

Übersetzen Sie die CSV-Datei und speichert sie in die Unterordner … \Debug

**Schritt 5:**

**LocBaml.exe /generate de-DE\UnisensViewer.resources.dll /trans:trans\_vi-VN.csv /out:vi-VN /cul:vi-VN**

**Schritt 6:**

**Thread.CurrentThread.CurrentCulture = new CultureInfo(“vi-VN“); Thread.CurrentThread.CurrentUICulture = new CultureInfo(“vi-VN“);**

## 