



.NET Internationalization

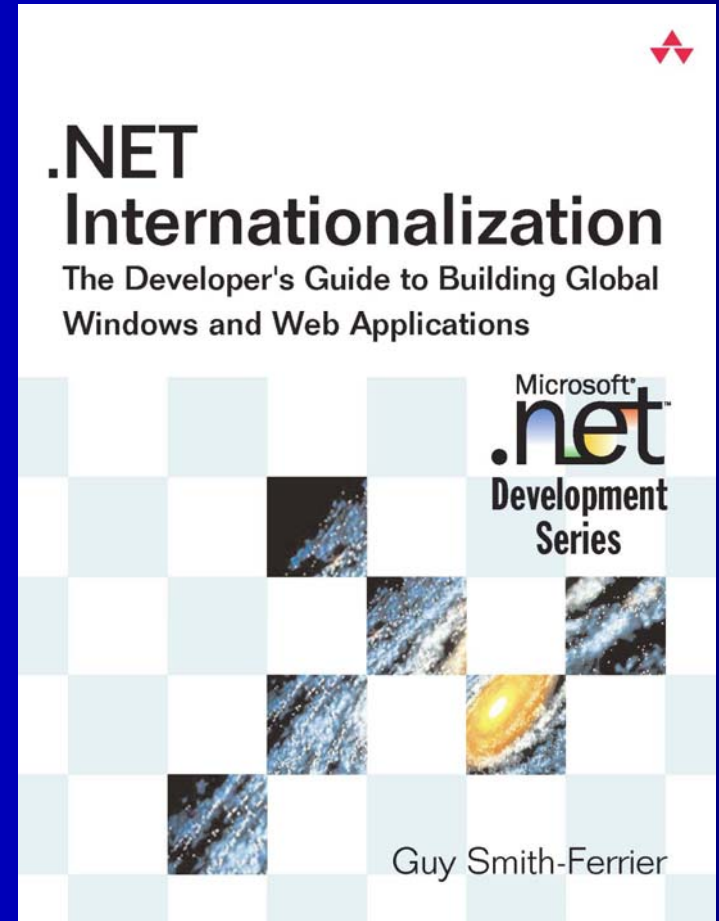


Guy Smith-Ferrier
guy@guysmithferrier.com

Blog: <http://www.guysmithferrier.com>

Author of...

- .NET Internationalization,
Addison Wesley,
ISBN 0321341384
– Due July 2006
- Visit
<http://www.dotneti18n.com>
to download the complete
source code

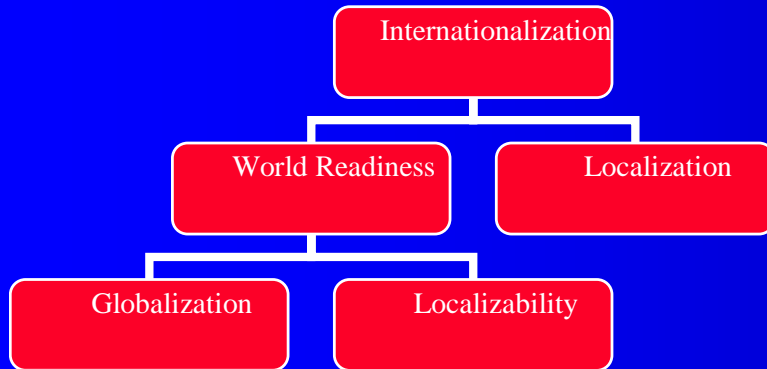


Agenda

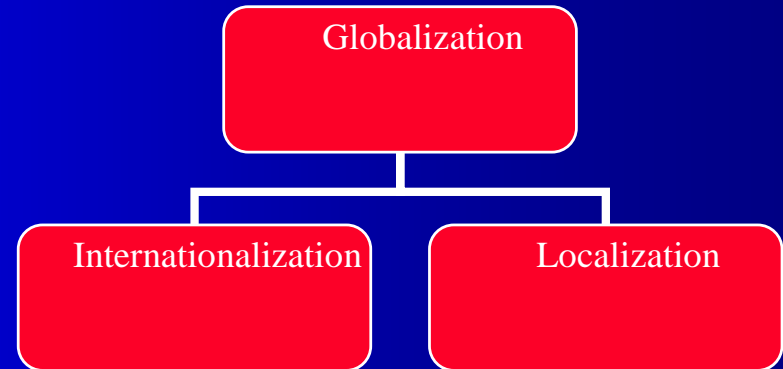
- Internationalization Fundamentals
 - Cultures
 - Localizing Strings
 - Resource Managers
 - Resource Assemblies
- Localizing Windows Forms Applications
- CurrentCulture and CurrentUICulture
- Localizing ASP.NET 2 Applications
- Globalization
- Strongly Typed Resources

Terminology Interpretations

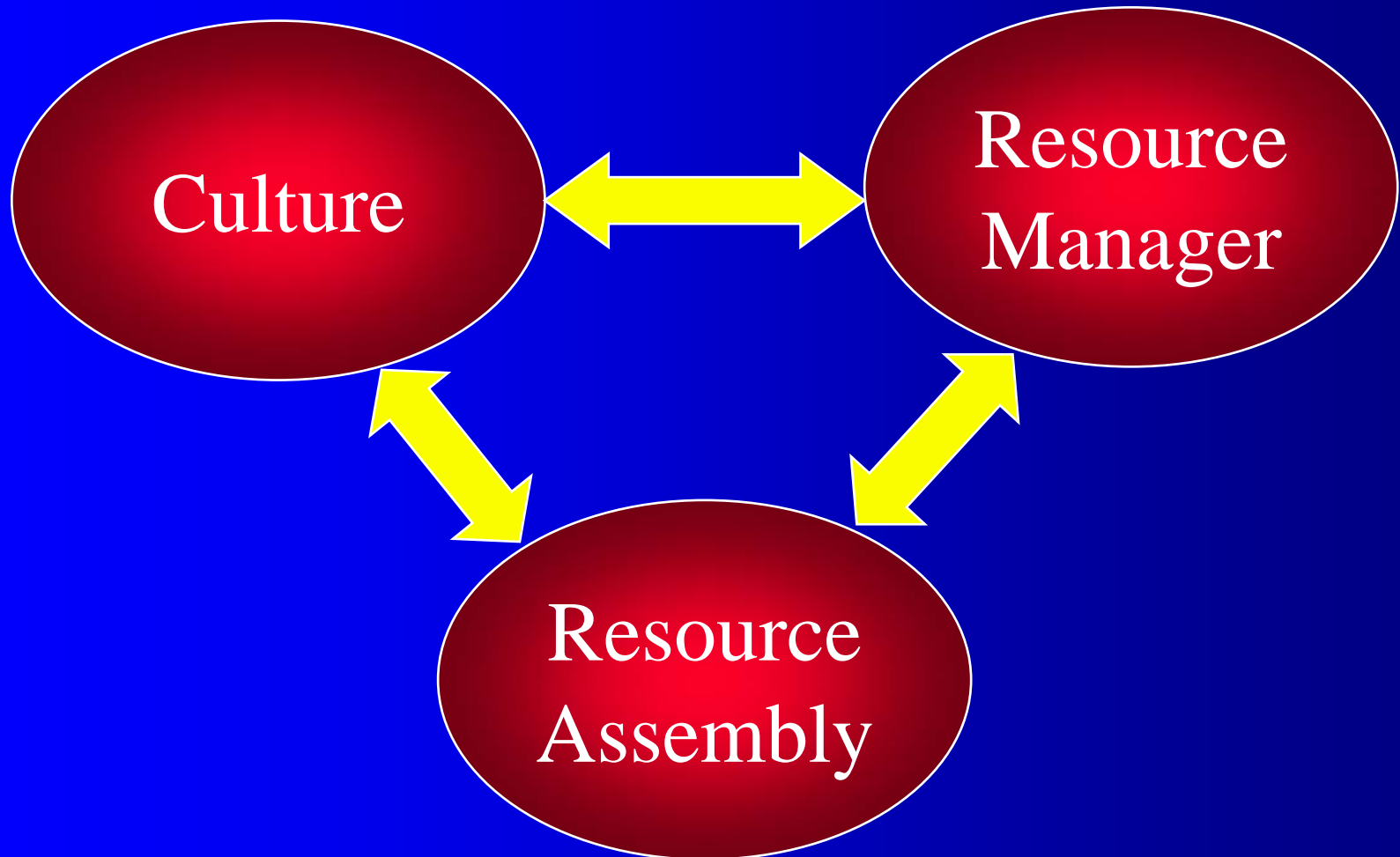
Microsoft Terminology



The Rest Of The Industry's Terminology



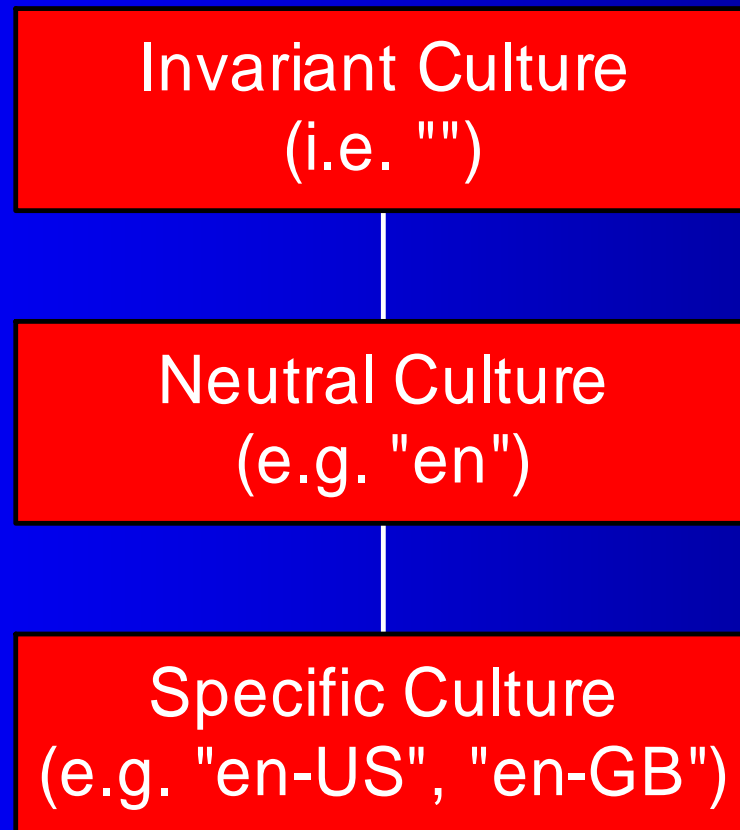
Internationalization Fundamentals



Culture Information

- Culture information is specified in the RFC 1766 format:-
 - languagecode2[-country/regioncode2[-script]]
 - languagecode2 is ISO 639-1 or 639-2
 - country/regioncode2 is ISO 3166
- For example:-
 - "en" (English, no specific country)
 - "en-US" (English in the US)
 - "en-GB" (English in the UK)
 - "en-AU" (English in Australia)
 - "fr" (French, no specific country)
 - "fr-FR" (French in France)
 - "fr-CA" (French in Canada)

Invariant Culture, Neutral Cultures and Specific Cultures



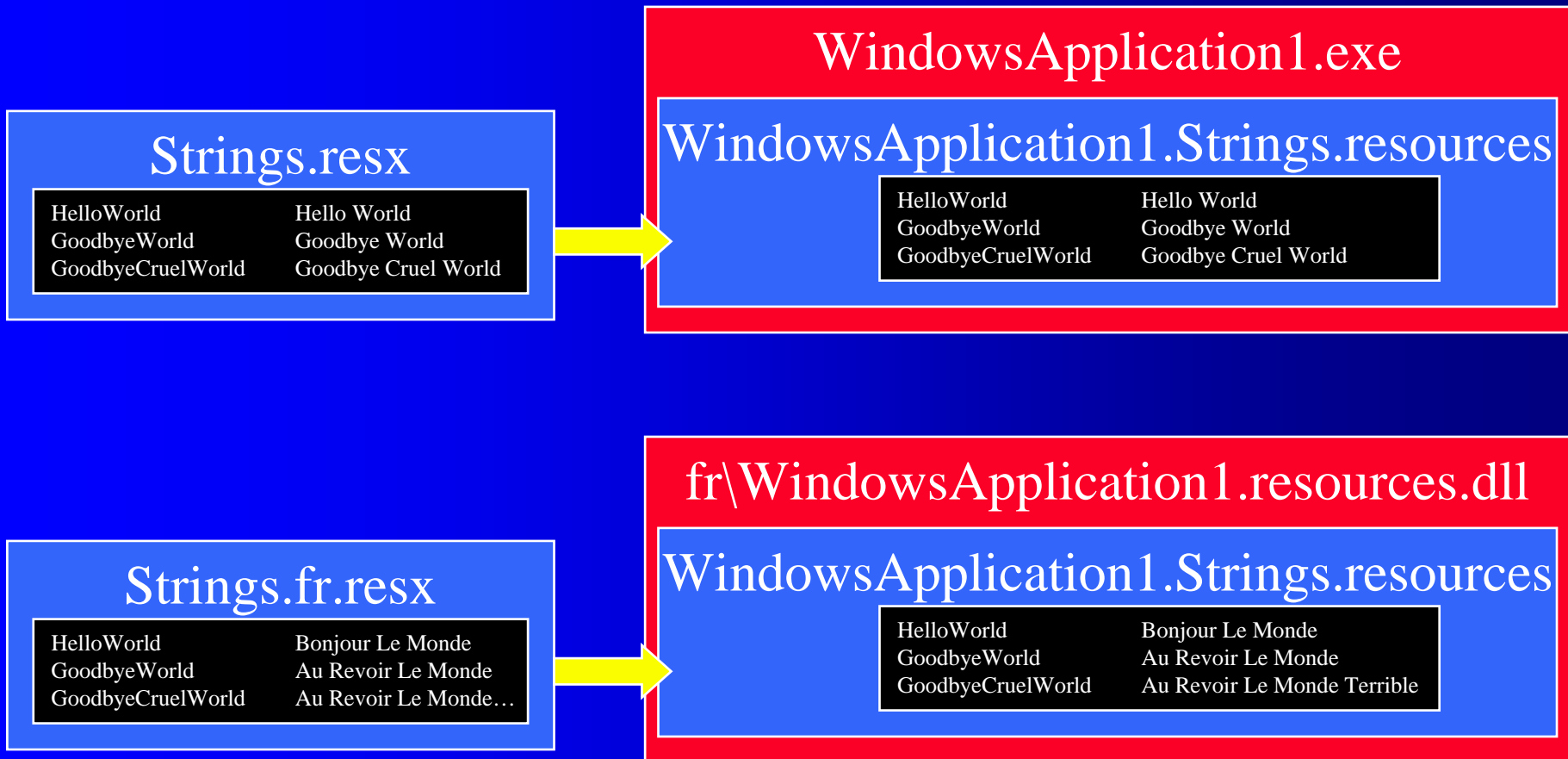
Demo

Localizable Strings

Demo

Localizing Strings

Resource Compilation



Demo

Localizable Windows Forms

Regional And Language Options

- From the Control Panel, select Regional and Language Options
- In the Regional Options page drop down the combo box and select a different language
- Click OK

CurrentCulture

- Culture information is set on a per thread basis
- `System.Threading.Thread.CurrentCulture`:-
 - Determines formats for dates, times, currency, and numbers, the sorting order of text, string comparisons, and casing
 - Does not determine the language setting
 - Defaults to `GetUserDefaultLCID (Win32)`
 - This value can be set in "Regional and Language Options"
 - `CultureInfo.CurrentCulture` is the same as `System.Threading.Thread.CurrentCulture` for the current thread
 - `CurrentCulture` must be a specific culture (e.g. "en-GB") and not a neutral culture (e.g. "en")
 - Use `CultureInfo.CreateSpecificCulture`

CurrentUICulture

- `System.Threading.Thread.CurrentUICulture`:-
 - Determines the language setting (because it is used by `ResourceManager`)
 - Defaults to `GetUserDefaultUILanguage (Win32)`
 - In Windows XP Multiple User Interface (MUI) this value can be set by the user
 - In all other versions of Windows it is determined by the language version of Windows which is installed
 - It cannot be changed by the user
 - `CultureInfo.CurrentUICulture` is the same as `System.Threading.Thread.CurrentUICulture` for the current thread
 - `CurrentUICulture` can be a specific culture (e.g. "en-GB") or a neutral culture (e.g. "en") or the invariant culture (i.e. "")

CurrentCulture and CurrentUICulture

- Culture information is set on a per thread basis
 - There is no built-in solution for changing the default culture information for new threads
 - It must be set manually

Demo

CurrentCulture and CurrentUICulture

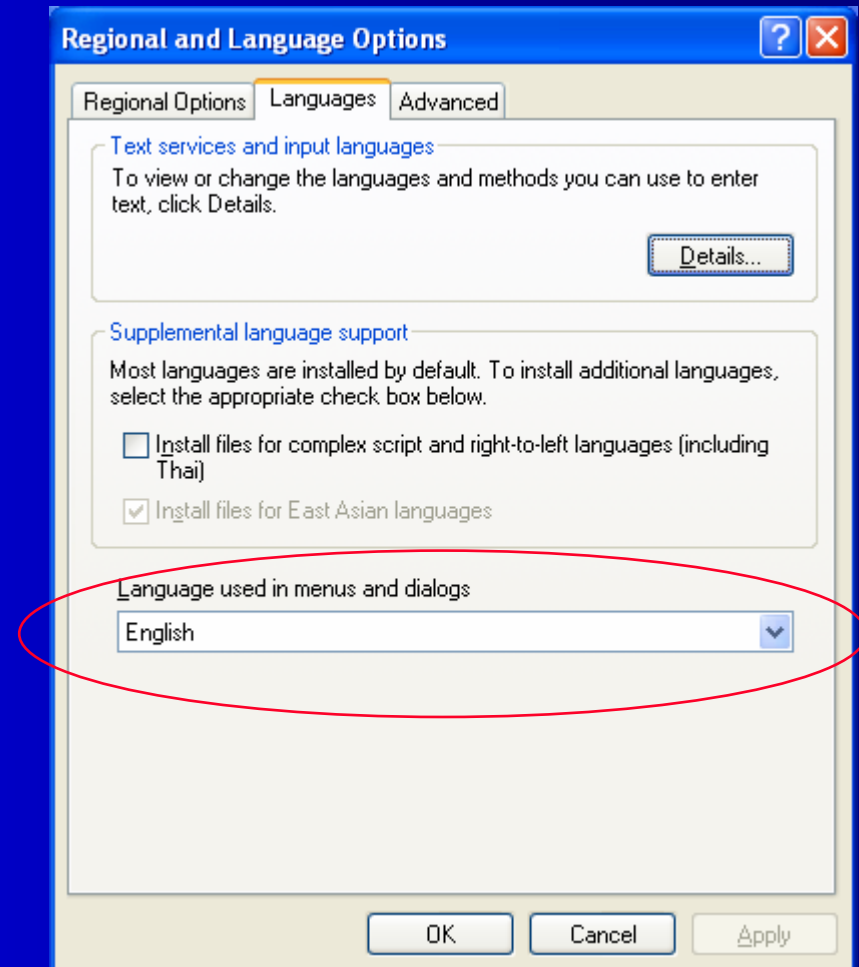
Changing The CurrentUICulture

- The default for the CurrentUICulture is determined by the version of Windows (for non-MUI Windows)
- To use the Regional Settings to change the language setting execute the following line at the beginning of the program and for each thread:-

```
Thread.CurrentThread.CurrentUICulture =  
Thread.CurrentThread.CurrentCulture;
```

Windows XP Multilingual User Interface (MUI) Pack

- Windows MUI is an user interface pack which can be installed on top of Windows XP
- Windows MUI adds a "Language used in menus and dialogs" combobox to the Regional and Language Options dialog



Localizability In Visual Studio 2005 ASP.NET

- Create a new ASP.NET application (File | New | Web Site...)

- Add a Label and look at the source code:-

```
<asp:Label ID="Label1" Runat="server" Text="Hello"  
></asp:Label>
```

- Select Tools | Generate Local Resource and look at the source code again:-

```
<asp:Label ID="Label1" Runat="server" Text="Hello"  
meta:resourcekey="LabelResource1">  
</asp:Label>
```

Demo

Localizability in Visual Studio 2005 ASP.NET Applications

Setting The Culture In ASP.NET

- You can set culture information in several ways:-

- 1. machine.config

```
<globalization culture="fr-FR" uiCulture="fr" />
```

- 2. web.config

```
<globalization culture="fr-FR" uiCulture="fr" />
```

- 3. In the Page:-

- In .NET 2.0 in the page header

```
UICulture="Auto" Culture="Auto"
```

- In .NET 1.1 and .NET 2.0 in the Page.Load event

```
Thread.CurrentThread.CurrentUICulture =  
new CultureInfo(Request.UserLanguages[0]);  
Thread.CurrentThread.CurrentCulture =  
CultureInfo.CreateSpecificCulture(Request.UserLanguages[0]);
```

Setting The Culture In Internet Explorer

- In Internet Explorer select Tools | Internet Options | General
 - Click the Languages... button
 - Click Add to add a new culture
 - You can also add "user defined" cultures
 - Click OK
 - Click OK

Globalization

- The `System.Globalization` namespace includes many classes which handle many globalization issues including:-
 - Date/time formats
 - Calendars
 - Number/currency formats
 - Casing
 - Sort orders

Strongly Typed Resources

- .NET 2.0 supports Strongly Typed Resources
 - A strongly typed resource is a class wrapper around a resource
 - It allows you to refer to a resource via a property of a class instead of by a string. Instead of writing:-
`resourceManager.GetString("HelloWorld")`
 - You write:-
`Strings.HelloWorld`
- Visual Studio 2005 automatically creates and maintains strongly typed resources for resx files
 - You can create strongly typed resources manually using `resgen`

Strongly Typed Resources Problems And Solutions

- There are two problems with strongly typed resources
 - resgen/ResXFileCodeGenerator only read from resx/resource files
 - If you use a different source for your resources then resgen/ResXFileCodeGenerator cannot read from your source
 - The generated code uses the ResourceManager class
 - If you use a different ResourceManager class the code is not usable
- The solution to both problems is that resgen/ResXFileCodeGenerator uses the StronglyTypedResourceBuilder class to create strongly typed resources
 - StronglyTypedResourceBuilder.Create accepts an IDictionary of resources
 - You can modify the CodeDom created by the Create method

Summary

- Internationalization Fundamentals
 - Cultures
 - Localizing Strings
 - Resource Managers
 - Resource Assemblies
- Localizing Windows Forms Applications
- CurrentCulture and CurrentUICulture
- Localizing ASP.NET 2 Applications
- Globalization
- Strongly Typed Resources