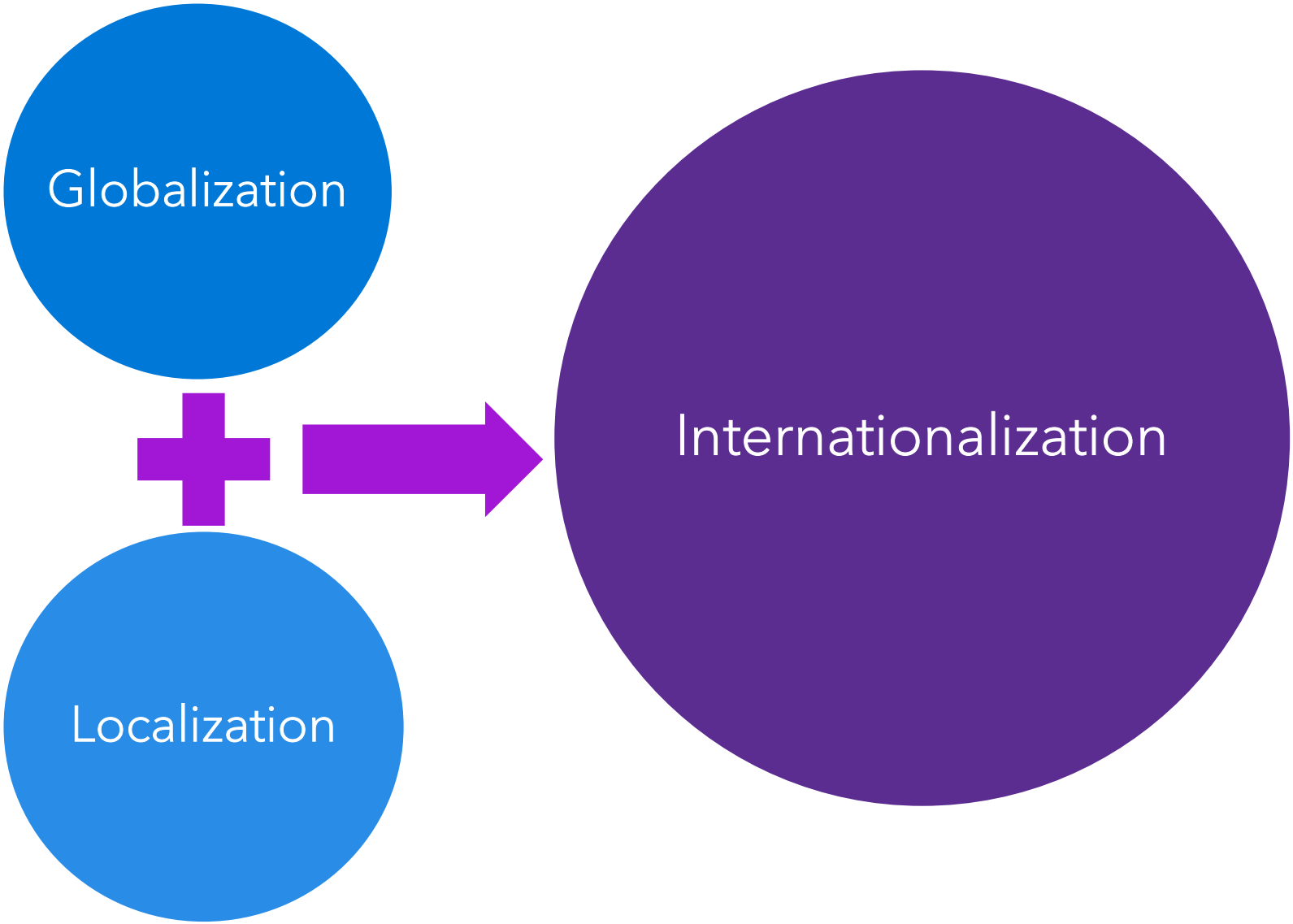
Globalizing Your ASP.NET 8 Presence

CONFOO 2024

Agenda

- Overview
- Globalizing ASP.NET Core MVC Projects
- Resource Files
- Localizing ASP.NET Core MVC Projects
- User Selectable Culture w/ Cookies
- Data Annotations
- Custom Providers
- Custom Localizers


Globalization
Localization
Internationalization

Globalization

Process of designing and developing software for different cultures

- Identifying the culture/locale that must be supported
- Designing features which support those cultures/locales
- Writing code that functions equally well in any of the supported cultures/locales

Globalization

Areas we have to look at for Globalization:

- Languages
- Currencies
- Date and time formatting
- Numeric, weight, and measure conventions

Localization

- Translating the application user interface (UI) or adapting graphics for a specific culture
- The localization process for assigning UI elements with localized text and resizing application UI elements to accommodate localized text

Globalization

Methods to Determine Culture

URL Request String

- `QueryStringRequestCultureProvider`

Cookies

- `CookieRequestCultureProvider`

Accept-Language HTTP Header

- `AcceptLanguageHeaderRequestCultureProvider`

Demo

Resource Files

Resource File Generation

`RESGEN.EXE`

Use a Localization Manager in your preferred IDE

Working with your Business Analyst

Localization:
Controllers

```
public class HomeController : Controller {
     private readonly IHtmlStringLocalizer<HomeController>
     _localizer;

     public HomeController(
          IStringLocalizer<HomeController> localizer
     )
     {
          _localizer = localizer;
     }
}
```

Resource File Naming Standard

Resources/Controllers.HomeController.en.resx
Resources/Controllers.HomeController.fr.resx
Resources/Controllers.HomeController.en.resx

Resources/Controllers/HomeController.en.resx
Resources/Controllers/HomeController.fr.resx
Resources/Controllers/HomeController.en.resx

Demo

Localization: Views

```
public IActionResult Index()
{
    ViewData["Title"] = _localizer["Title"];
    return View();
}
```

```
@USING MICROSOFT.ASPNETCORE.MVC.LOCALIZATION

@INJECT IVIEWLOCALIZER LOCALIZER

@{
    VIEWDATA["TITLE"] = LOCALIZER["HOMETITLE"];
}
<H1 CLASS="PAGE-HEADER">@VIEWDATA["HOMETITLE"]
</H1>
```

Resource File Naming Standard

Resources/Views.Home.Index.en.resx

Resources/Views.Home.Index.fr.resx

Resources/Views.Home.Index.en.resx

Resources/Views/Home/Index.en.resx

Resources/Views/Home/Index.fr.resx

Resources/Views/Home/Index.en.resx

Demo

Localization: Data Annotation

Resource File Naming Standard

Models/AlbumViewModel.en-US.resx

Models/ AlbumViewModel.en-US.resx

Models/ AlbumViewModel.en-US.resx

Demo

Localization: Custom Localizer

Demo

Thanks for attending the session!

All the demoes and code located on GitHub

https://github.com/cwoodruff/ASPNETCore-I18n