|  |  |
| --- | --- |
|  | **Unidev** |
|  | **Unified Development, Inc.**  16690 Swingley Ridge Road  Suite 260  Chesterfield, MO 63017  (636) 532-4424 |

|  |
| --- |
|  |
|  |

**Duncan Solutions, Inc**.

PEMS Localization

November 2013

Revision 1.00

**Prepared by:**

Ron Howard

Caleb Miller

Unified Development

**Table of Contents**

[1. Overview 4](#_Toc373947747)

[1.1 Goal 4](#_Toc373947748)

[2. Target Audience 5](#_Toc373947749)

[3. Localization 6](#_Toc373947750)

[3.1 PEMS Controller 6](#_Toc373947751)

[3.1.1 Configure Localization Data 6](#_Toc373947752)

[3.2 Client: Using Localization 7](#_Toc373947753)

[3.3 Server: Using Localization 7](#_Toc373947754)

[4. Globalization / Custom Labels 8](#_Toc373947755)

[4.1 Globalization 8](#_Toc373947756)

[4.1.1 Extended Resource Provider 8](#_Toc373947757)

[4.1.2 Automatic Inserts 9](#_Toc373947758)

[4.1.3 Adding a new locale 9](#_Toc373947759)

[4.2 Custom Labels 9](#_Toc373947760)

[4.2.1 Locale Resources Custom 9](#_Toc373947761)

[5. Time Zones 12](#_Toc373947762)

**Revision** History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Author | Description | Revision Date |
| 1.00 | Caleb Miller | Initial version | 12/4/13 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Overview

## Goal

The purpose of this document is to explain the concepts and methodologies used when localizing the site. There are two main concepts this document will cover:

1. Localization: the process of manipulating the dates and times displayed to the user to conform to the specific customer in which they are logged into.
   1. Explanation of how the application determines what times and dates to display
   2. How the application acts on those values from the Client and Server
2. Globalization
   1. Resource provider - Locale Resources and how the application globalizes the terms used on the site
   2. Custom Labels – how to allow each customer to have term specific overrides for the locale resources

# Target Audience

The target audience of this document is person or persons who have:

1. Experience in the following technologies:
   1. C# /.Net / MVC
   2. Visual Studio 2012
   3. SQL Server Management Studio
2. The user of this document has a full understanding of the Duncan PEMS project. This includes database and system architecture knowledge, desired business rules of the application, etc.
3. Basic knowledge of how globalization works with resource (.res) files in the .Net environment.
4. Microsoft SQL Server administration and understand rights, database creation and administration, and are able to use either SSMS or SQL command line interface. The user will need the ability to update table data.

# Localization

The PEMS application needs to localize dates and times displayed to the user based on the context of the customer they are logged in as. If the user is sitting in the United States, but is viewing the North Sydney Council site, then the application needs to override the default culture being used (en-US) and use the culture set for North Sydney Council (en-AU).

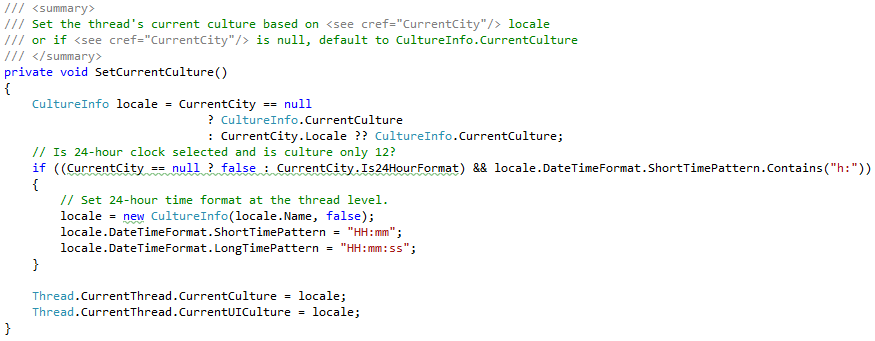
This is done by setting the culture to the culture specific for that customer. There are two main areas that need localization: Client side and Server side. Each of these components has to render the correct date time format and a 24hour clock for the times if that is required for that customer. This 24 hour flag is set on the customer in the Duncan Administration application as well for each specific customer. Each customer also has a specific time zone, so this needs to be taken into account when displaying the local time or a customer.

## PEMS Controller

A base controller exists that overrides the OnActionExecuting event that all authenticated controllers need to use: /Duncan.PEMS.Framework/Controller/PemsController.cs

This controller takes the Executing context, determines the current customer from the route data passed in and sets View Data variables for the Client and Session variables for the Server.

### Configure Localization Data

1. Culture: Sets the .net Current Culture to the culture of the customer. This allows .Net to have the correct culture set for use on the server side. 
2. The base controller also sets various Controller.ViewData of the executing context and Session information to provide access to this data on the client side. The pertinent values for this document are: Local time, Current Locale, and 24 hour format.





By using the PEMS Controller, every View / Action / Controller using it has access to the appropriate culture, 24 hour clock settings, and current local time for the customer. Below are the specifics on how each main area uses these settings.

## Client: Using Localization

Typically the users’ browser determines the culture it uses based on auto-detection of the location of the user. The browsers default settings needs to be updated with the correct culture information for the current customer.

All of the controls that display date time to the user are Kendo controls. This is a requirement of the system to function correctly. In order to localize the entire client side of the site, the Kendo culture is set on the master page. As long as the application uses the Kendo controls to display date times to the user, then it will be localized correctly. These items are set in the \_Layout.cshtml file that is used throughout the application and is located: /Areas/shared/Views/Shared/\_Layout.cshtml. All of the user authenticated pages in the application use this layout, so it is set here and it is persisted to all the views in the site. This is the code snippet inside of the <head> tag of the \_Layout file:



This allows us to assign a date time to a kendo control and it will render in the correct culture specific format.

## Server: Using Localization

Localizing dates and times on the server requires no additional effort, as the culture is set on the thread by the base controller. This allows us to use DateTime.Now.ToShortDateString (), etc. to render the date and time in the correct format. To view the customers’ current local time, retrieve the associated Session data that was set by the PEMS Controller.

# Globalization / Custom Labels

Duncan needed the ability to have text displayed to the user that was specific to the language of the current customer. On top of that, each customer needs the ability to override certain globalized terms. These terms are modified by using the applications GUI within the administration section of the site. This means globalization within the PEMS system falls into two main categories: Globalization and customer specific custom labels.

## Globalization

Normal resource files used for globalization are set at compile time and cannot be changed without resetting the application. The PEMS system needs the ability to change resource files while the application is running, so regular .Net globalization cannot be used in this system.

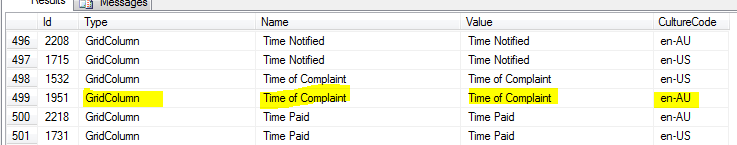
### Extended Resource Provider

To meet this requirement, the following MSDN article was followed to extend the default Resource-Provider model to use a Database resource provider:

<http://msdn.microsoft.com/en-us/library/aa905797.aspx> (See “Building a Database Resource Provider” section). The implementation resides in the Duncan.PEMS.Utilities project in a class file called SqlExtendedResourceProvider.cs. The application uses this process to display all the terms on the site to the user. Here is an example of usage, which is used throughout the site:

<h2>@HttpContext.GetGlobalResourceObject( ResourceTypes.GridColumn, "Time of Complaint" )</h2>

What this does is pull data from the DB for the Type (Grid Column) and Name (Time of Complaint) for the culture defined for the current customer (which is set on the current thread). The name of the table all of the data is stored in is “Locale Resources”. Here is a snapshot of some of the data:



Items to note on how the extended resource provider works:

1. When initially retrieving data, all the items of the type for that culture are cached, not just the term requested. So the first call to the GetGlobalResourceObject with the ResourceTypes.GridColumn type, all of the Grid Columns for that culture are retrieved once and cached. Any following calls check to see if the term requested resides within the cache, and if not, re-builds the cache for that type and culture.
2. This cache is flushed when the application is reset.

### Automatic Inserts

The resource provider was extended resource provider to automatically insert data if it is not found in the DB. If this is added to a view on the site:

@HttpContext.GetGlobalResourceObject( ResourceTypes.GridColumn, "Testing" )

The application would check to see if the term exists for the current culture, and if not, insert that data with the Value of the term the same as the Name passed in (“Testing” in this example). This allows development to go much quicker since the application takes care of inserting new terms; all the developer has to do is use the method. If a user logs in as a customer with a culture that does not exist in the system, these assets will automatically get generated as the user navigates throughout the site.

### Adding a new locale

This example will use the Spanish – Argentina locale (es-AR). There are two ways a locale can be added to the system.

#### Excel Import

The first is to export all of the en-US resources to an excel file, update the “Value” of each term, and set the culture code to es-AR, then use import tool in SQL Server Management Studio (SSMS) to import the data into the Local Resources Table. This is the preferred method for adding new cultures to the Local Resources in the system.

#### Manual Updates

The second is to create a customer and assign them the es-AR culture, then login to the customer and navigate the site. This will generate all of the terms for the culture automatically, but these terms will be using the Name field as the Value. In order to update these values, a user will have to either user SSMS to edit the data or a command line interface to update the Value field for each associated term in the system for the new culture (es-AR).

This process will also need to be used when a new term is added to the system. If the “Testing” term above is added, this term will only reside in the DB once the user navigates to a page that uses the same Type and will only be created for the culture that is assigned to the current customer they are logged into. So, to add “Testing” to the es-AR culture:

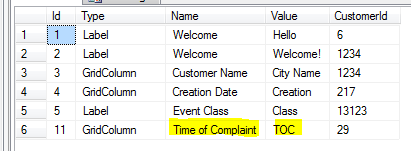
1. Manually go in and do and insert the data via SSMS
2. OR Navigate to the page while logged in as our es-AR customer, then update the “Value” field of the appropriate table row that was created.

## Custom Labels

Along with culture specific terms, the application also needs the ability to customize each term on a per customer basis. The extended resource provider is limited to a single culture and is not customer specific. In order to accommodate this requirement, customized locale resources have been implemented.

### Locale Resources Custom

A table has been added to store all of the customer specific labels and grid columns. These custom terms only apply to locale resources with the Type of “Grid Column” or “Label”. All other term types are not customer specific. Here is a snapshot of the data from the Locale Resources Custom table:



#### Using custom resources

This section will use the term “Time of Complaint” as an example. In order to get this system to work correctly and allow customers to have term specific values, the extended resource provider has been modified to perform the following logic:

When a term is requested, this custom table is checked to see if the term exists for the current customer. If it exists, then what has been cached for that term is ignored and the customer specific value is used instead. If a custom term does NOT exist for this customer, then the normal process for local resources I followed.

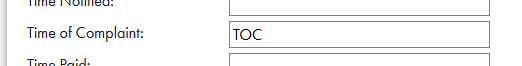
This allows the application to have customer specific terms on a case by case basis with a fallback to the Culture specific term. In our Time of Complaint Example, any time the term is requested for customer 29, TOC is displayed instead of what has been cached for the Time of Complaint term.

#### Setting Custom Resources

The application provides the ability to modify the custom labels on a per customer basis. This user interface resides in the Duncan Administration customer under the Administration > Client Management > Customer Details > Custom Labels tab



This page lists all of the local resources that are marked as Grid Column or Label. It allows the user to add the custom labels they require.



Saving this page will save the data into the Locale Resources Custom table.

# Time Zones