## EnumConvert

各Domain枚举类型约定与其多语言方案和实现：

1. **枚举类型放置统一地方：** 放在Service端的ECCentral.BizEntity.Enum中,每个Domain会有一个对应的cs文件，放这个Domain的所有定义的enum（注意是业务相关的enum）。

在Portal端也会有有ECCentral.BizEntity.Enum，里面都是从Service端link方式添加的相应文件

1. **枚举类型多语言方案：**在定义枚举类型是请为枚举类型指定Description属性来指定多语言类型（如CustomerType的[Description("ECCentral.BizEntity.Resources.ResCustomerEnum")，如果没有[Description]属性则会用默认的资源类"ECCentral.BizEntity.Resources.ResCommonEnum" , Description参数为资源类全名。），在对应的多语言文件中定义的Key为“[枚举类型名称]\_[枚举值名称]”如（“CustomerType\_Campus”）
2. **枚举Converter 类**：**ECCentral**.Portal.Common.Utilities.EnumConverter
3. **提供取得列表的静态方法：**
4. ECCentral.Portal.Common.Utilities.EnumConverter. GetKeyValuePairs<TEnum>(EnumAppendItemType appendType)：返回值为 **List<KeyValuePair<Nullable<TEnum>, string>>**
5. ECCentral.Portal.Common.Utilities.EnumConverter. GetKeyValuePairs<TEnum>() **=** ECCentral.Portal.Common.Utilities.EnumConverter. GetKeyValuePairs<TEnum>(EnumAppendItemType.None)

**参数说明： appendType：**EnumAppendItemType.None表示不插入。

* + 1. EnumAppendItemType.All表示插入默认的“所有”项到返回列表中，且插入到第一个位置。Key值为null，Value值为：ECCentral.BizEntity.Resources. ResCommonEnum.Enum\_All
    2. EnumAppendItemType.Select表示插入默认的“请选择”项到返回列表中，且插入到第一个位置。Key值为null，Value值为：ECCentral.BizEntity.Resources. ResCommonEnum.Enum\_Select
    3. EnumAppendItemType.Custom\_All表示插入自定义的“所有”项到返回列表中，且插入到第一个位置。Key值为null，Value值为：[资源类（由枚举类型的Description属性定义的资源类）].[枚举类型名称]，如果没有定义资源则用ECCentral.BizEntity.Resources. ResCommonEnum.Enum\_All
    4. EnumAppendItemType.Custom\_Select表示插入自定义的“请选择”项到返回列表中，且插入到第一个位置。Key值为null，Value值为：[资源类（由枚举类型的Description属性定义的资源类）].[枚举类型名称\_\_Select] （**注：为发不在资源不重复Key所以为**“枚举类型名称\_\_Select”），如果没有定义资源则用ECCentral.BizEntity.Resources. ResCommonEnum.Enum\_Select

**注意**：插入附加项的 Key值为null。

1. **取得枚举值对应的描述的方法:**
2. 外部方法如果: Enum.ToDescription() ，**由**ECCentral.Portal.Common.Utilities.EnumExtension.ToDescription(this Enum value)定义
3. ECCentral.Portal.Common.Utilities. EnumConverter. GetDescription(Enum value);
4. ECCentral.Portal.Common.Utilities. EnumConverter. GetDescription(object enumValue, Type enumType);
5. **SilverLight使用方式举例**
6. 下拉列表绑定：

.xaml：<ComboBox x:Name="cmbCustomerRank" DisplayMemberPath="Value" SelectedValuePath="Key" SelectedValue="{Binding [TEnum:字段名称]}">

.xaml.cs：cmbCustomerRank.ItemsSource = EnumConverter.GetKeyValuePairs<CustomerRank>(EnumConverter.EnumAppendItemType.All)

1. 枚举值页面显示：

.xaml：<TextBlock x:Name="tbkCustomerRank"/>

.xaml.cs：tbkCustomerRank.Text = Enum.ToDescription()

1. DataGrid列表绑定：

.xaml：

<ovsDataControls:DataGrid>

<ovsDataControls:DataGrid.Columns>

<ovsDataControls:DataGridTextColumn ovsDataControls:DataGridAttached.Header="{Binding Path=DataGrid\_Column\_Head\_SOStatus ,

Source={StaticResource MyResSO}}" Binding="{Binding Status, Converter={StaticResource MyEnumConverter}}" CanUserSort="False" />

</ovsDataControls:DataGrid.Columns>

</ovsDataControls:DataGrid>

**注：**

1. 取得枚举描述时以枚举类型上通过DescriptionAttribute配置的资源最为优先，如果没有配置DescriptionAttribute将会自动去找类名为“ECCentral.BizEntity.Resources.ResCommonEnum”的资源。
2. 在资源文件中以“[枚举类型名称]\_ [枚举值名称]”为Key（如：“CustomerType\_Campus”）取得枚举描述。