**Command parameters**

**Definice uplne normalne:**

public ICommand ChangeStatusCommand { get; }

ChangeStatusCommand = new RelayCommand(DoChangeStatus);

**Zmena je v obsluzne metode:**

Parametr dostaneme jako object, je potreba ho pretypovat.

private void DoChangeStatus(object obj)

{

var statusType = (byte) (PublicationStatusValues) obj;

}

**Xaml:**

Je potreba predat parametr:

<MenuItem

Header="{x:Static gui:GuiResources.NotSupplied}"

Visibility="{Binding Path=SelectedPublication.PublicationStatusType,

Converter={StaticResource PublicationStatusConverter},

ConverterParameter={x:Static global:PublicationStatusType.NotSupplied}}"

Command="{Binding Path=ChangeStatusCommand}"

CommandParameter="{x:Static enums:PublicationStatusValues.NotSupplied}" /> // enumy jsou staticke a jsou videt vsude

# **Jak predat UIElement jako parametr do ViewModelu**

**UIElement dostaneme do VM pomoci jeho Name jako parametr commandu**

<telerik:RadGridView

Name="RadGridView" // Nazev elementu ktery chceme dostat do VM

<Button

Grid.Column="1"

Width="100"

Margin="2,0"

Command="{Binding Path=ExportCommand}"

CommandParameter="{Binding ElementName=RadGridView}"

Content="{x:Static gui:GuiResources.Export}"

IsEnabled="{Binding Path=ExportCanExecute}" />

public ICommand ExportCommand { get; }

ExportCommand = new RelayCommand(DoExport);

private void DoExport(object obj) //**parametr je object**

{

var radGrid = (RadGridView) obj; **// nutne pretypovani**

const string extension = "xls";

var dialog = new SaveFileDialog

{

DefaultExt = extension,

Filter = string.Format("{1} files (.{0})|.{0}|All files (.)|.", extension, "Excel"),

FilterIndex = 1

};

if (dialog.ShowDialog() == true)

{

try

{

using (var stream = dialog.OpenFile())

{

// RadGridView.FilterDescriptors.Count

radGrid.Export(stream,

new GridViewExportOptions

{

Format = ExportFormat.ExcelML,

ShowColumnHeaders = true,

ShowColumnFooters = true,

ShowGroupFooters = false

});

}

}

catch (Exception exception)

{

ShowExportError(exception);

}

}

}

# **Jak obslouzit event UIElementu , predaneho jako parametr commandu**

<telerik:RadGridView Name="PricedItems" ContextMenu="{StaticResource baseMenu}" >

<telerik:RadGridView.InputBindings>

<KeyBinding Key="LeftCtrl" Modifiers="Control" Command="{Binding Path= SetClipboardCommand}" CommandParameter="{Binding ElementName=PricedItems}"/>

</telerik:RadGridView.InputBindings>

InputBindigs ma byt na elementu ktery predavam. Na usercontrolu mi to nefungovalo , obj byl null

public ICommand SetClipboardCommand { get; }

SetClipboardCommand = new RelayCommand(DoRegisterListener);

private void DoRegisterListener(object obj)

{

var radgrid = (RadGridView) obj;

if (radgrid != null)

{

radgrid.Copied += SetClipboardDataFormat;

}

}

private void SetClipboardDataFormat(object sender, RadRoutedEventArgs e)

{

var dataObj = Clipboard.GetDataObject();

var data = dataObj?.GetData(DataFormats.Text);

if (data != null)

{

Clipboard.SetData(DataFormats.Text, data);

}

var radgrid = (RadGridView)sender;

radgrid.Copied -= SetClipboardDataFormat;

}