using System;

using System.Linq;

using DevExpress.ExpressApp;

using DevExpress.ExpressApp.Editors;

using DevExpress.ExpressApp.Model.Core;

using DevExpress.XtraTreeList.Nodes;

using Xafari.BC.Settings;

using Xafari.BC.Settings.Runtime;

using Xafari.BC.Settings.Win;

namespace Xafari.Northwind.Win

{

 public class HideSettingsViewController : ViewController<DetailView>

 {

 public HideSettingsViewController()

 {

 this.TargetViewId = "SettingsObject\_DetailView\_User";

 }

 protected override void OnActivated()

 {

 base.OnActivated();

 TryInitializeAnniversaryItem();

 }

 public void TryInitializeAnniversaryItem()

 {

 PropertyEditor propertyEditor = ((DetailView)View).FindItem("Keys") as PropertyEditor;

 if (propertyEditor != null)

 {

 if (propertyEditor.Control != null)

 {

 SubscribeToBeforeExpandEvent(propertyEditor);

 }

 else

 {

 propertyEditor.ControlCreated += new EventHandler<EventArgs>(propertyEditor\_ControlCreated);

 }

 }

 }

 private void SubscribeToBeforeExpandEvent(PropertyEditor propertyEditor)

 {

 ((WinSettingsPropertyEditor)propertyEditor).\_groupTreeList.BeforeExpand += \_groupTreeList\_BeforeExpand;

 }

 public SettingsObject CurrentSettingsObject

 {

 get { return this.View.CurrentObject as SettingsObject; }

 }

 void \_groupTreeList\_BeforeExpand(object sender, DevExpress.XtraTreeList.BeforeExpandEventArgs e)

 {

 CheckNodes(e.Node);

 }

 private void CheckNodes(TreeListNode parentNode)

 {

 if (parentNode.Nodes.Count == 0) return;

 var groups = string.IsNullOrEmpty((string)parentNode.Tag)

 ? this.CurrentSettingsObject.ActualSettings as ModelNode

 : this.CurrentSettingsObject.ActualSettings.FindGroupNodeByPath((string)parentNode.Tag) as ModelNode;

 if (groups == null) return;

 for (var i = 0; i < groups.NodeCount; i++)

 {

 var modelNode = groups.GetNode(i) as Model.Product;

 if (modelNode != null)

 {

 var node = parentNode.Nodes.FirstOrDefault(x => (string)x.Tag == ((IModelBCSettingsGroup)modelNode).SettingsPath);

 if (node != null) node.Visible = false;

 }

 }

 }

 private void propertyEditor\_ControlCreated(Object sender, EventArgs e)

 {

 SubscribeToBeforeExpandEvent((PropertyEditor)sender);

 var editor = sender as WinSettingsPropertyEditor;

 if (editor == null) return;

 var rootNode = editor.\_groupTreeList.Nodes[0];

 CheckNodes(rootNode);

 }

 protected override void OnDeactivated()

 {

 base.OnDeactivated();

 }

 }

}