

Using the Hierarchy Navigator Control for Dependent Select Lists

Document ID: Q000064

Last Revised On: Wednesday, August 08, 2007

This article applies to the following:

Component(s):

Administrator

Solutions(s):

All

Summary

A commonly requested feature for issue tracking forms is select list dependencies in which a selection in one list determines the possible values in another, and so on. Dependent select lists are useful whenever a user needs to categorize an item according to rules that organize the terms in a hierarchy of dependent values. For example, one list would allow user to categorize a problem as one relating to either "Hardware" or "Software", and then based on that selection, another list would display additional category values based on the selection in the first list. This tech tip describes how to use the Hierarchy Navigator control to add dependent select lists to IssueNet rich clients, such as the IssueNet Manager or IssueNet Workspace, as well as browser based products like IssueNet Relay.

Using the Hierarchy Navigator Control in Rich Client Forms

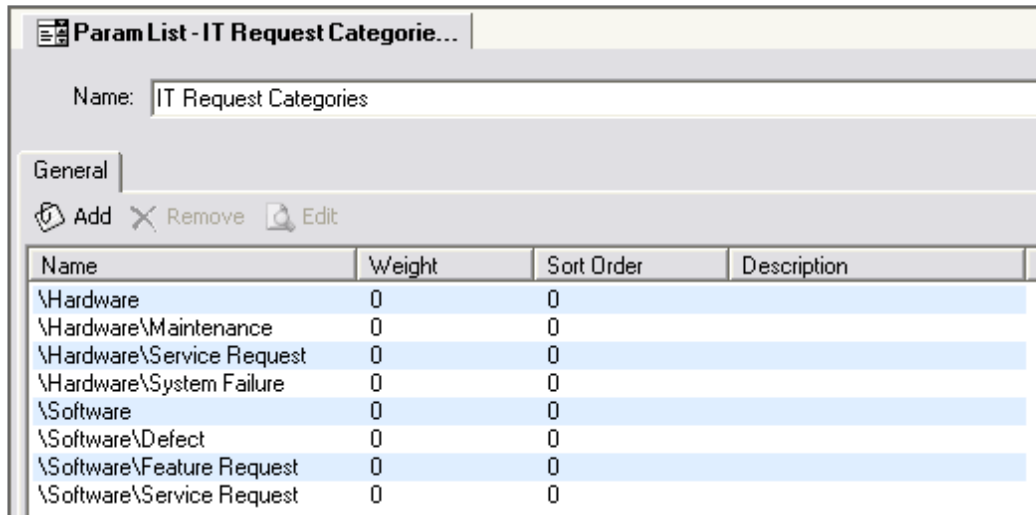
Adding the Select List Values

The first step in using the hierarchy navigator control is to create a parameter list which defines the values which will be displayed in the select lists loaded by the control. The Hierarchy Navigator control works by taking the individual parameter list values you define and breaking them down into sub-values based on delimiters you add to the each parameter list item. Consider the example in the summary of this tech tip. The relationships between the values in a primary select list with the values "Hardware" and "Software" and sub-categories dependent upon those selections could be expressed as follows.

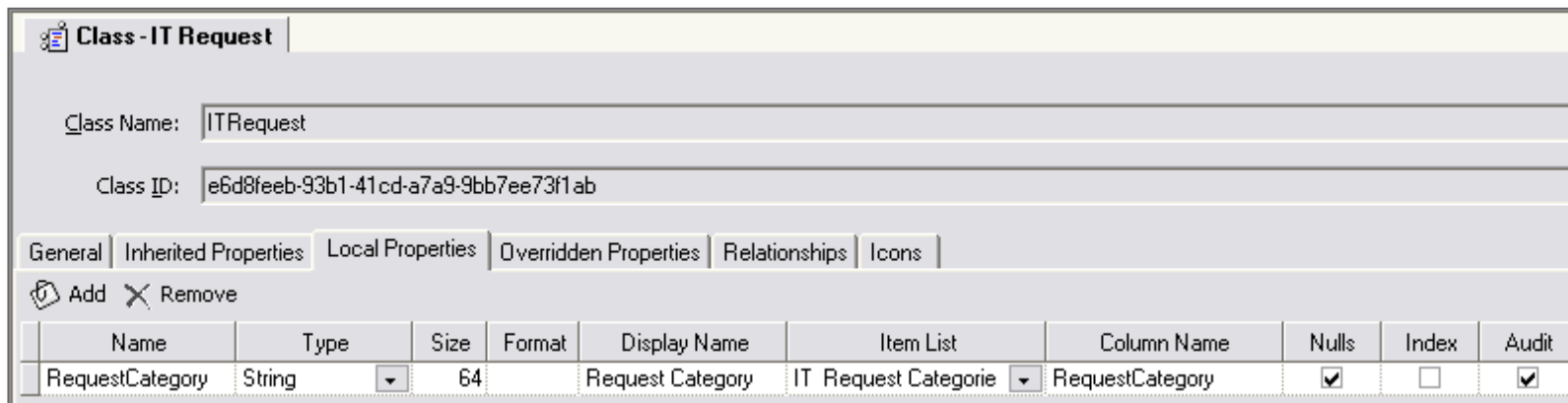
```
\Software
\Software\Defect
\Software\Feature Request
\Software\Service Request
\Hardware
\Hardware\Maintenance
\Hardware\System Failure
\Hardware\Service Request
```

By adding delimiters to the parameter list, the list can define the values for each select list the hierarchy navigator control will display and their dependencies. This method of defining the values also makes it easy to modify and maintain the values since it only requires you to edit a single parameter list.

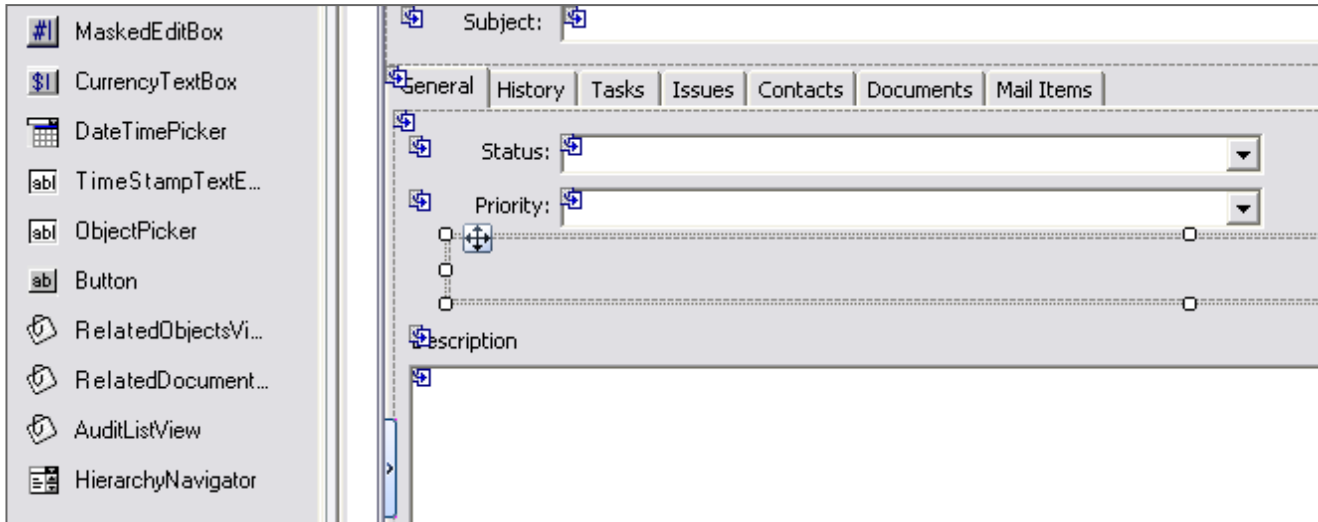
To get started, launch the Architect and add a new parameter list like the one illustrated in the following screen capture. Note that the control will allow you to use any character you like as your delimiter value. However, you will want to choose a character that will not be used as a character in any of the select list values. You can also nest the values as deeply as you like, the navigator control will interpret the delimiters to create as many select lists as necessary.



Once you have created the parameter list, you will want to bind the list to a string property for the class you want to use the new list with. The following screen capture illustrates the new item list bound to a string property.



Once you have added the parameter list and bound it to a class property, you are ready to add the hierarchy navigator control to one of your forms. Open the Architect and open the form to which you want to add the hierarchy navigator. In the Form Toolbox click on the Hierarchy Navigator control and then click and drag on the form to add the control to the form. Drag out the control dimensions so that it will be large enough to accommodate the number of select lists the control will create based on number of levels in the parameter list.



Once you have added the control, you will want to specify the following properties in the Form Properties pane to indicate how the control will load the value from the parameter list:

[-] Appearance	
BackColor	<input type="text" value="Control"/>
Error on errorProvider	
[+] Font	
Font	Tahoma, 8.25pt
ForeColor	<input type="text" value="ControlText"/>
IconAlignment on errorProv	MiddleRight
IconPadding on errorProvi	0
[-] Behavior	
Enabled	True
TabIndex	12
TabStop	False
Visible	True
[-] Data	
MemberName	RequestCategory
[-] Design	
(Name)	ITRequestCategory
GenerateMember	True
Modifiers	Private
[-] Focus	
CausesValidation	True
[-] Layout	
Anchor	Top, Left
Dock	None
[+] Location	
Location	87, 95
[+] Size	
Size	684, 32
[-] Misc	
EmptyItemText	[None]
HierarchyDelimiter	\
Levels	(Collection) <input type="button" value="..."/>
StartWithDelimiter	True

MemberName

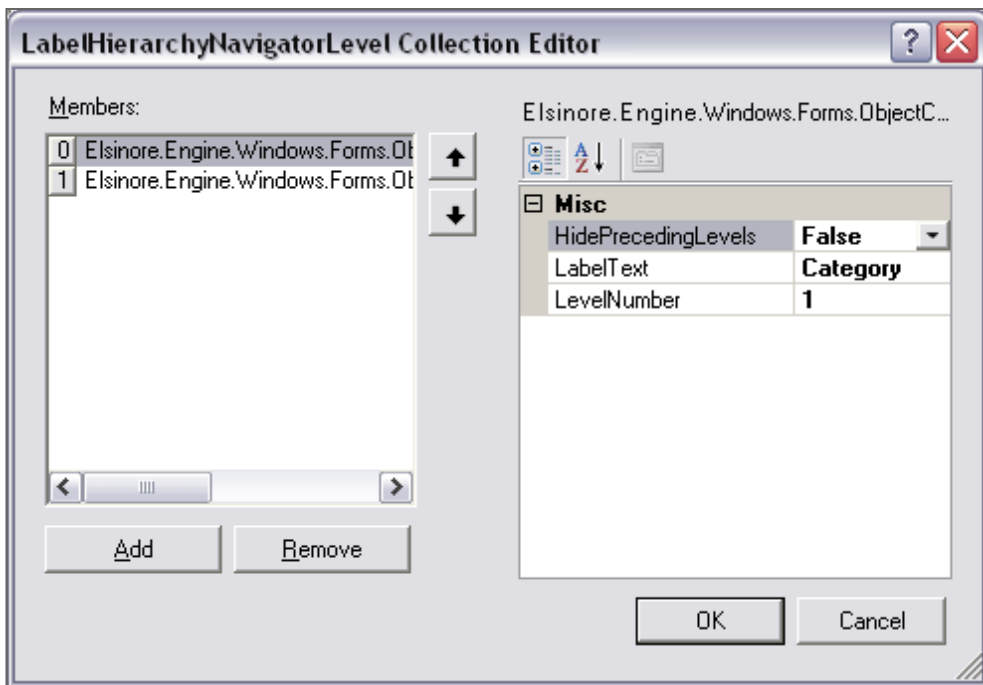
For the **MemberName** property select the name of the class property you linked the parameter list to.

HierarchyDelimiter

For the **HierarchyDelimiter** property, specify the character you used to delimit the different levels in the parameter list. In this example, the delimiter used is the backslash “\”.

Levels

Click on the ellipsis button in the row for the Levels property to open the levels editor. The levels editor allows you to define the properties for each level, or select list, the control will display. The **HidePrecedingLevels** property determines if the control will ignore levels in the control prior to the current one. The **LabelText** property sets the value which will be used to label the select list for each level in the hierarchy. The **LevelNumber** property determines the order of the levels from left to right in the control. Note: when setting level numbers you will need to set a value greater than 0.



StartWithDelimiter

The **StartWithDelimiter** property allows you to specify whether each of the parameter list values begins with a delimiter. In this example all of our parameter values begin with the backslash “\”. Accordingly, this property is set to true.

The Results

Once you have specified all of the properties mentioned above, you are ready to make any final adjustments and view the form. For information on other form and control properties not specific to the Hierarchy Navigator, please review the Elsinore tech tip on [Modifying IssueNet Forms](#). The following screen capture displays the results of our example. Note: when this form is in use the second select list will not appear until a value is selected in the first.

General | History | Tasks | Issues | Contacts | Documents | Mail Items

Status:

Priority:

Category: Sub-Category:

Description:

[None]
Maintenance
Service Request
System Failure

Using the Hierarchy Navigator Control in IssueNet Relay

The process for adding a Hierarchy Navigator control to a Relay form is similar to adding one to an IssueNet rich client form. The parameter list values and the class property are specified in the Architect exactly as they were in the previous example. The difference in adding the control to a Relay form is that, because Relay does not feature a drag and drop forms editor, the settings for the control need to be added as tags in the form page the class is using. If you logon to Relay as the Administrator and open the Administration section, you can click on the Classes tab and view the name of each page used by each class. You can then navigate to the Issue Pages directory, e.g., C:\Program Files\Elsinore Technologies\IssueNet\Web Applications\Relay\IssuePages, and open the page for the form you want to modify in the editor of your choice. Once you have opened the page, choose where you want to add the Hierarchy Navigator on the form and add the following code to the page with the values for the control properties set to match your configuration. The following example has its values set to load the same property as the rich client example and load the values in two select lists. The names of control properties you will want to modify in the following code are the same as the ones modified for the rich client control. In the following code you will want to set the following values exactly as described in the previous example:

- HierarchyDelimiter – determines the character the control will use as a delimiter
- MemberName – determines the class property the control is bound to
- LevelNumber – determines the order of each level displayed by the control
- HidePrecedingLevels – determines if the preceding level in the hierarchy is hidden
- Text – determines the label text for each level in the hierarchy

```
<elo:HierarchyNavigator runat="server" HierarchyDelimiter="\ " ID="RequestCategory"
  MemberName=" RequestCategory " MemberBinding="Value" AutoPostBack="true">
  <elo:TemplateHierarchyNavigatorLevel LevelNumber="1" HidePrecedingLevels="false">
    <Template>
      <tr>
```

```
<td class="FieldLabelCell">
    <els:Label runat="server" Text="Category:" />
</td>
<td class="FieldValueCell">
    <els:DropDownList runat="server" ID="ListControl" />
</td>
<td class="FieldMiscellaneousCell">
</td>
</tr>
</Template>
</elo:TemplateHierarchyNavigatorLevel>
<elo:TemplateHierarchyNavigatorLevel LevelNumber="2" HidePrecedingLevels="false">
    <Template>
        <tr>
            <td class="FieldLabelCell">
                <els:Label runat="server" Text="Sub-Category:" />
            </td>
            <td class="FieldValueCell">
                <els:DropDownList runat="server" ID="ListControl" />
            </td>
            <td class="FieldMiscellaneousCell">
            </td>
        </tr>
    </Template>
</elo:TemplateHierarchyNavigatorLevel>
</elo:HierarchyNavigator>
```

Conclusion

The Hierarchy Navigator control provides a simple and consistent way to add a variety of linked select lists to IssueNet forms for both rich and browser based clients. By allowing you define the value relationships into a single parameter listing using delimiters, modifying and maintaining the list values remains simple without the need to a separate configuration files or scripts.
