

Custom Widget: TreeView

Contents

Introduction	1
Interfaces	1
Properties:.....	1
Events:.....	2
Remarks	2
Building a TreeView	2
Remarks:	4
Map of Revisions.....	4

Introduction

The TreeView custom widget allows the user to build and manipulate different nodes through a hierarchical graphical interface.

Interfaces

Properties:

Property Name	Action	Description
Format	Set	This property defines the format of the data handled by the custom widget. The following values are supported: 0 (default): CSV 1: JSON
ConfigurationFile	Set	When this property changes of value, it will load the configuration settings from the respective file name to build the TreeView nodes.
ConfigurationText	Set	When this property changes of value, it will build the TreeView nodes according to this property value.
SelectedNodeSettings	Get	Returns all settings associated with the currently selected node.
SelectedNode	Get+Set	Returns the path and name of the selected node. When this property changes, it automatically selects the node defined in the value of this property (if it exists) on the tree view.
SelectedPath	Get	Returns the path of the selected node
SelectedLabel	Get	Returns the label of the selected node

SelectedData	Get	Returns the data of the selected node
SelectedIcon	Get	Returns the icon of the selected node
ExpandAll	Set	When this property changes of value, it will expand all nodes in the TreeView.
ExpandNode	Set	When this property changes of value, it will expand the currently selected node.
CollapseAll	Set	When this property changes of value, it will collapse all nodes in the TreeView.
CollapseNode	Set	When this property changes of value, it will collapse the currently selected node.

Events:

Event	Description
onSelectedChanged	This event is triggered when the selected node changes. By the time that the event is triggered, the user can get information about the selected node through the respective properties (SelectedNodeSettings, SelectedNode, etc).
onExpandAll	This event is triggered when all nodes are expanded.
onExpandNode	This event is triggered when a single node is expanded.
onCollapseAll	This event is triggered when all nodes are collapsed.
onCollapseNode	This event is triggered when a single node is collapsed.

Remarks

Building a TreeView

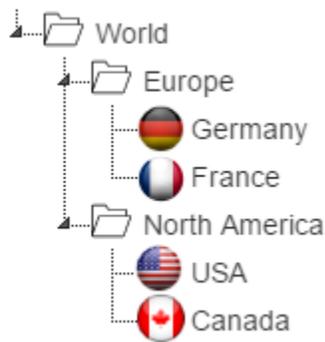
In order to build the TreeView, the user can configure the settings either through the ConfigurationFile property or through the ConfigurationText property. Even though both properties can be configured, it seldom be a valid configuration from a practical point of view, since the last one modified will overwrite the settings from the previous one.

The configuration will follow the following syntax, depending on the value of the Format property:

Format Value	Syntax	Example
0 (csv)	<node label>,<node path>,<node data>,<node icon> <node label>,<node path>,<node data>,<node icon>	World Europe,World Germany, World\Europe,DE,FlagGE France,World\Europe, FR,FlagFR North America,World USA,World\North America,EN,FlagUS Canada,World\North America, EN;FR,FlagCA

1 (json)	<pre>{ "label": "<node label>" , "data": "<node data>" , "icon": "<node icon>" , "children": [{ "label": "<node label>" , "data": "<node data>" , "icon": "<node icon>" , "children": [{ "label": "<node label>" , "data": "<node data>" , "icon": "<node icon>" }] }] }</pre>	<pre>{ "label": "World", "data": "", "icon": "", "children": [{ "label": "Europe", "data": "", "icon": "", "children": [{ "label": "Germany", "data": "DE", "icon": "FlagGE" }, { "label": "France", "data": "FR", "icon": "FlagFR" }] }, { "label": "North America", "data": "", "icon": "", "children": [{ "label": "USA", "data": "EN", "icon": "FlagUS" }, { "label": "Canada", "data": "EN;FR", "icon": "FlagCA" }]} }</pre>
----------	--	---

The graphical representation of the TreeView described in the previous example should look like the following picture:



Remarks:

- The syntax is the same both for the content of the file pointed by the “ConfigurationFile” property as well as for the actual value of the “ConfigurationText” property.
- When using the “ConfigurationText” property in the CSV format, the separator between two nodes is a CR and LF (ASCII 13 and 10) characters.
- If the “ConfigurationFile” property extension is omitted, it should be considered “.csv” or “.json” by default, depending on the value of the “Format” property. Optionally, the user can customize the file extension in the value of the “ConfigurationFile” property.
- If the path is omitted for the “ConfigurationFile” property, the configuration file must be found in the custom widget root directory. Optionally, the “ConfigurationFile” property can support a relative path from the custom widget root directory.
- The only mandatory value in the syntax for one node is the <node label>.
- If the <node path> is left in blank, the node is added to the root.
- If the <node icon> is left in blank, the default icon (illustrated in the root directory of the previous picture) will be displayed. The default icon will automatically change to an open or close folder, based on the node expansion state.
- If the <node icon> extension is omitted, it should be considered “.ico” by default.
- The <node icon> should support at least “.ico” and “.png” pictures. The picture should be resized automatically to fit the area available in the TreeView.
- If the path is omitted in the <node icon> value, the picture file must be found in the custom widget root directory. Optionally, the <node icon> can support a relative path from the custom widget root directory.
- The format of the value returned by the “SelectedNodeSettings” property depends on the value of the “Format” property (CSV or JSON). The returned value should not include any CR and LF (ASCII 13 and 10) characters.
 - o If Format=0 (csv): <node label>,<node path>,<node data>,<node icon>
 - o If Format=1 (json): {"label":"<node label>","path":"<node path>","data":"<node data>","icon":"<node icon>"}
- The object supports ASCII characters only (not UNICODE).

Map of Revisions

Revision	Date	Author	Description
A	November 3 rd , 2017	Fabio Terezinho	Initial Revision
B	November 21 st , 2017	Fabio Terezinho	Removed the requirements to support UNICODE characters due to technical constraints