

17.9.0 TreeView

Das Steuerelement `TreeView` implementiert eine Baum-Ansicht einer Struktur.

Die Klasse `_TreeView` ist die interne Basis-Klasse für das Steuerelement `TreeView`, aber auch für die Steuerelemente `ListView` und `ColumnView`. `TreeView` ist nicht mehr in `gb.qt4`, sondern in `gb.gui.base` implementiert. Die Dokumentation unter → <http://gambaswiki.org/wiki/comp/gb.qt4/treeview> listet das `TreeView` trotzdem in `gb.qt4` auf, weil `gb.gui.base` eine versteckte Komponente ist, die Steuerelemente sammelt, die in Gambas geschrieben werden können und zwischen `gb.gtk*` und `gb.qt*` geteilt werden sollen. `TreeView` ist in Gambas geschrieben und deshalb hat man die Möglichkeit den Quelltext des Steuerelements einzusehen.

- Die Elemente der Baum-Ansicht werden durch einen Schlüssel indiziert. Sie zeigen einen Text und optional ein Symbol für jedes Element. Jeder Eintrag bekommt einen eindeutigen Schlüssel, wenn er zur `TreeView` hinzugefügt wird.
- Das Steuerelement verfügt über einen internen Zeiger. Verwenden Sie die unterschiedlichen `Move`-Methoden, um den internen Zeiger zu bewegen und die Item-Eigenschaft, um auf das aktuelle Element zuzugreifen.
- Wenn Sie den internen Zeiger auf ein bestimmtes Element setzen wollen, dann müssen Sie die Methode `MoveTo(Key)` benutzen.
- Um den den sichtbaren Zeiger – als Auswahl eines bestimmten Eintrags – zu setzen, weisen Sie der `Key`-Eigenschaft eines Elements der `TreeView` den Schlüssel des zu markierenden Eintrags zu.

Mit den verschiedenen `Move`-Methoden können Sie den internen Zeiger verschieben. `MoveParent()` zum Beispiel setzt den internen Zeiger auf sein `Parent`-Element – wenn es ein `Parent`-Element gibt. Beachten Sie, dass diese `Move`-Methoden den Wert `True` zurück geben, wenn die Bewegung des internen Zeigers *nicht* ausgeführt werden konnte, weil:

- es kein `Parent`-Element gibt (bei `MoveParent`),
- es kein nächstes Element gibt (bei `MoveNext`) oder
- der interne Zeiger kein `Kind`-Element besitzt (zum Beispiel `MoveChild`).

17.9.0.1 Eigenschaften

Die Klasse `TreeView` verfügt u.a. über diese Eigenschaften:

Eigenschaft	Datentyp	Beschreibung
<code>Current</code>	<code>_TreeView_Item</code>	Gibt das aktuelle Element zurück. Im Allgemeinen ist es das Element, das den Fokus hat oder das markiert ist (Single-Selection-Modus).
<code>Item</code>	<code>_TreeView_Item</code>	Gibt das Element zurück, auf das der interne Zeiger zeigt oder <code>NULL</code> , wenn der interne Zeiger nicht erreichbar ist.
<code>Count</code>	<code>Integer</code>	Gibt die Anzahl aller Elemente in der <code>TreeView</code> zurück.
<code>Editable</code>	<code>Boolean</code>	Gibt an, ob die Elemente standardmäßig editierbar sind, das heißt, ob sie vom Benutzer umbenannt werden können, wenn er auf sie klickt.
<code>Key</code>	<code>String</code>	Gibt den <code>Key</code> des aktuellen Elements zurück.
<code>Mode</code>	<code>Integer</code>	Setzt den Auswahlmodus oder gibt den Auswahlmodus zurück. Sie finden in der <code>Select</code> -Klasse eine Liste der Konstanten, die Sie mit dieser Eigenschaft verwenden können.
<code>Selection</code>	<code>String[]</code>	Zurückgegeben werden die <code>Keys</code> aller markierten Elemente in einem <code>String-Array</code> (Multi-Selection-Modus).
<code>Sorted</code>	<code>Boolean</code>	Zeigt an, ob die Elemente in der <code>TreeView</code> sortiert angezeigt werden.

Tabelle 17.9.0.1.1 : Ausgewählte Eigenschaften der Klasse `TreeView`

Im nächsten Abschnitt werden Ihnen die wichtigsten Eigenschaften und Methoden der virtuellen Klasse `_TreeView_Item` vorgestellt, da die Eigenschaften `Current` und `Item` vom Typ `_TreeView_Item` sind.

17.9.0.2 Eigenschaften und Methoden der Klasse `_TreeView_Item`

Die virtuelle Klasse `_TreeView_Item` repräsentiert einen Eintrag in einer TreeView.

17.9.0.3 Eigenschaften der Klasse `_TreeView_Item`

Eigenschaft	Datentyp	Beschreibung
Children	Integer	Gibt die Anzahl der untergeordneten Elemente zu einem Element zurück.
Count	Integer	Gibt die Anzahl der untergeordneten Elemente zu einem Element zurück.
Editable	Boolean	Gibt an, ob dieses Element editierbar ist. Das bedeutet, ob es vom Anwender umbenannt werden kann, wenn er darauf klickt.
Expanded	Boolean	Ermittelt oder legt fest, ob das Element ausgeklappt ist, das heißt, ob seine Child-Elemente zu sehen sind.
Key	String	Gibt den Key zurück.
ParentKey	String	Gibt den Key des Parent-Elements zurück oder NULL, wenn das Element kein Parent-Element besitzt.
Picture	Picture	Gibt das Bild zurück oder setzt das Bild, das neben dem Element angezeigt wird.
RichText	String	Gibt den RichText zurück oder setzt den RichText.
Selected	Boolean	Gibt an, ob das Element ausgewählt ist.
Text	String	Gibt den angezeigten Text zurück oder setzt den anzuzeigenden Text.

Tabelle 17.9.0.3.1 : Ausgewählte Eigenschaften der virtuellen Klasse `_TreeView_Item`

17.9.0.4 Methoden der Klasse `_TreeView_Item`

Methode	Rückgabotyp	Beschreibung
Clear()	-	Löscht alle Child-Elemente
Delete()	-	Löscht das (aktuelle) Element.
MoveAfter ([Key As String])	-	Bewegt das Element direkt hinter das Element, dessen Schlüssel als Parameter angegeben ist. Die beiden Elemente müssen das gleiche Parent-Element haben. Wenn der optionale Parameter 'Key' nicht angegeben ist, so wird das Element das erste Child-Element im Parent-Element.
MoveBefore ([Key As String])	-	Bewegt das Element direkt vor das Element, dessen Schlüssel als Parameter angegeben ist. Die beiden Elemente müssen das gleiche Parent-Element haben. Wenn der optionale Parameter 'Key' nicht angegeben ist, so wird das Element das <i>letzte</i> Child-Element im Parent-Element.
MoveFirst()	-	Verschiebt das Element, so dass es das erste Child-Element im Parent-Element wird.
MoveLast()	-	Verschiebt das Element, so dass es das letzte Child-Element im Parent-Element wird.
Rename()	-	Startet das Umbenennen des Elements.

Tabelle 17.9.0.4.1 : Ausgewählte Methoden der virtuellen Klasse `_TreeView_Item`

17.9.0.5 Methoden

Die Klasse `TreeView` besitzt u.a. diese Methoden:

Methode	Rückgabotyp	Beschreibung
Exist (Key As String)	Boolean	Zurückgegeben wird der Wert True, wenn das Element mit dem angegebenen Key existiert.

Methode	Rückgabotyp	Beschreibung
Add (Key As String, Text As String [, Picture As Picture, Parent As String, After As String])	_TreeView_Item	Fügt ein Element in die TreeView ein.
MoveBack()	Boolean	Wenn eine der anderen Move-Methoden versagt, so können Sie diese Methode benutzen, um den internen Zeiger zurück an seine Position vor dem Umzug zu bewegen. Es wird True zurück gegeben, wenn der interne Zeiger nicht auf einer gültigen Position war.
MoveBelow()	Boolean	Bewegt den internen Zeiger auf ein sichtbares Element unterhalb des aktuellen. Gibt True zurück, wenn unten kein Element existiert.
MoveChild()	Boolean	Bewegt den internen Zeiger auf das erste untergeordnete Element. Gibt True zurück, wenn kein Element existiert.
MoveCurrent()	Boolean	Bewegt den internen Zeiger auf das aktuelle Element. Es ist das Element, das den Fokus hat oder das markiert ist. Gibt True zurück, wenn kein Element existiert.
MoveFirst()	Boolean	Bewegt den internen Zeiger auf das erste Element in der TreeView. Gibt True zurück, wenn die TreeView leer ist.
MoveLast()	Boolean	Bewegt den internen Zeiger auf das letzte Element in der TreeView. Gibt True zurück, wenn die TreeView leer ist.
MoveNext()	Boolean	Bewegt den internen Zeiger auf das nächste Kind-Element. Gibt True zurück, wenn es kein Kind-Element gibt.
MovePrevious()	Boolean	Bewegt den internen Zeiger zum vorherige Kind-Element. Gibt True zurück, wenn es kein Kind-Element gibt.
MoveParent()	Boolean	Bewegt den internen Zeiger auf das übergeordnete Element. Gibt True zurück, wenn es kein übergeordnetes Element gibt.
MoveTo (Key As String)	Boolean	Bewegt den internen Zeiger auf ein bestimmtes Element. Gibt True zurück, wenn das Element zu dem Key nicht existiert.

Tabelle 17.9.0.5.1 : Ausgewählte Methoden der Klasse TreeView

Die Add(...)-Methode fügt ein neues Element in die TreeView ein:

```
Function Add ( Key As String, Text As String [ , Picture As Picture, Parent As String, After As String ] )
    As _TreeView_Item
```

- *Key* ist der Schlüssel für das neue Element.
- *Text* ist der angezeigte Text auf dem neuen Element.
- *Picture* ist das Bild, das neben dem Text angezeigt wird. Standardmäßig wird kein Bild angezeigt.
- *Parent* ist der Schlüssel des übergeordneten Elements, in das das Element eingefügt werden soll. Standardmäßig ist das neue Element ein Root-Element.
- *After* ist der Schlüssel von dem Element, nach dem das neue Element eingefügt wird. Dieses Element und das neue Element müssen das gleiche Parent-Element haben. Standardmäßig wird das neue Element als letztes Element des Parent-Elements eingefügt.

Beispiel aus dem Projekt POP3-Client → Kapitel 24.5.4:

```
For Each sElement In aAccounts
    TreeView1.Add(sElement, sElement, Picture["icon:/32/mail"])
    TreeView1.Add(sElement & "SMTP", "SMTP", Picture["Icons/smtp.png"], sElement)
    TreeView1.Add(sElement & "POP3", "POP3", Picture["Icons/pop3.png"], sElement)
Next
```

Für drei EMail-Konten ergibt sich dann die folgende Ansicht in der TreeView:

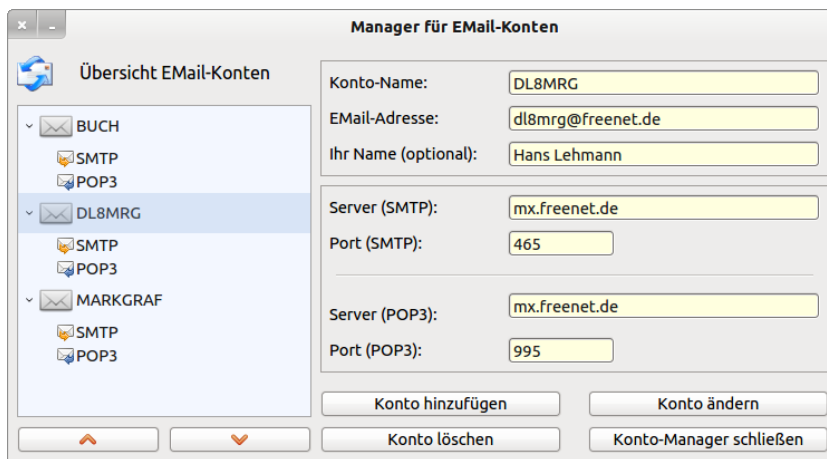


Abbildung 17.9.0.5.1: TreeView mit markiertem Eintrag

17.9.0.6 Ereignisse

Die Klasse *TreeView* besitzt ferner die folgenden Ereignisse:

Ereignis	Beschreibung
Click()	Das Ereignis wird ausgelöst, wenn auf ein Element in der TreeView geklickt wird.
Select()	Das Ereignis wird ausgelöst, wenn sich die Auswahl in der TreeView ändert. Beachten Sie: Click() wird bei jedem Klick auf ein Element in der TreeView geklickt ausgelöst – also zum Beispiel auch dann, wenn man nur einen Eintrag auf- oder einklappen will. Select() dagegen wird nur dann ausgelöst, wenn sich die Auswahl ändert.
Compare (Key As String, OtherKey As String)	Dieses Ereignis wird ausgelöst, wenn zwei Elemente in der TreeView verglichen werden. Das Ergebnis des Vergleichs wird in der <i>Compare</i> -Eigenschaft gespeichert.

Tabelle 17.9.0.6.1 : Ausgewählte Ereignisse der Klasse TreeView

Eine Anwendung der Klasse *TreeView* nutzen Sie immer dann ausführlich, wenn Sie die IDE von Gambas starten. Dort wird die physische Struktur des Projekt-Verzeichnisses auf eine logische Struktur in einer TreeView abgebildet:

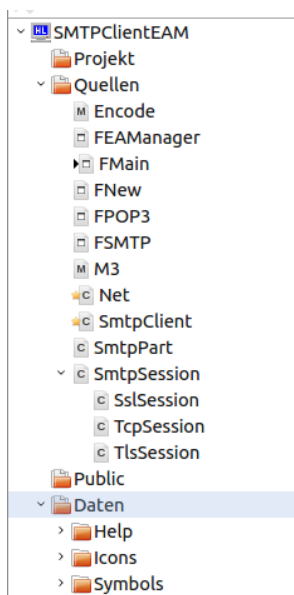


Abbildung 17.9.0.6.1: TreeView in der Gambas-IDE