

## 17.6 IconView

Die Komponente IconView (gb.qt4) implementiert ein Steuerelement, das beschriftete Symbole anzeigen kann. IconView-Elemente werden durch einen Schlüssel indiziert. Die Komponente verfügt über einen internen Cursor für den Zugriff auf seine Elemente. Verwenden Sie u.a. die Methoden MoveFirst, MoveNext, MovePrevious, MoveBack oder MoveTo, um den *internen* Cursor zu bewegen und die *Item*-Eigenschaft, um das aktuelle Element anzuzeigen.



Abbildung 17.6.1: IconView im Projekt Explorer

Das Projekt *Explorer* in den Beispielen von Gambas unter der Rubrik *Verschiedenes* nutzt eine IconView für die Anzeige der Verzeichnisstruktur.

### 17.6.1 Eigenschaften und Methoden IconView

In den folgenden 2 Tabellen werden ausgewählte Eigenschaften und Methoden einer IconView beschrieben:

Eigenschaft	Typ	Beschreibung
Ascending	Boolean	Zeigt an, ob die Elemente in einer IconView aufsteigend sortiert werden oder nicht (→ absteigend).
Compare	Integer	Verwenden Sie diese Eigenschaft, um das Ergebnis eines Vergleichs im Compare-Ereignis zurückzugeben (0, 1 oder -1).
Count	Integer	Gibt die Anzahl der Elemente in der IconView an.
Current	IconView.Item	Liefert das aktuelle, ausgewählte Element in der IconView.
Editable	Boolean	Gibt an, ob die Elemente standardmäßig editierbar sind, wenn ein Element ausgewählt wurde. Mit der Funktionstaste <i>F2</i> wird der Edit-Modus gestartet und mit <i>Enter</i> der geänderte Text übernommen.
GridSize	Integer	Gibt die Gittergröße der IconView zurück oder setzt den Wert.
IconLines	Integer	Gibt die maximale Anzahl der Zeilen zurück, die mit einem Symbol geschrieben werden oder setzt den Wert.
IconSize	Integer	Gibt die Größe des größten Symbols in der IconView zurück.
Key	String	Gibt den Key-String des aktuellen Elements in der IconView zurück.
Mode	Integer	Ermittelt den Selektions-Modus oder legt ihn fest (Konstanten: Select.None (0), Select.Single (1) oder Select.Multiple (2)).
Orientation	Integer	Gibt die Ausrichtung des angezeigten Textes und des Icon zurück oder setzt den Wert mit <i>Arrange.Vertical</i> oder <i>Arrange.Horizontal</i> .
Sorted	Boolean	Legt fest, ob die Elemente in der IconView sortiert <u>angezeigt</u> werden oder nicht.
Picture	Picture	Gibt das Bild zurück, das als <i>Hintergrundbild</i> der IconView festgelegt wurde oder setzt das Bild.
Renaming	Boolean	Gibt True zurück, wenn ein Element der IconView umbenannt wurde.

Tabelle 17.6.1.1: Ausgewählte Eigenschaften der Komponente IconView

Sie können einer IconView ein selbst definiertes Kontext-Menü zuordnen, das Sie über einen Klick mit der rechten Maustaste (RMT) aufrufen.

Die Ausrichtung des angezeigten Textes und des Icon kann die Werte *Arrange.Vertical* (Text unter dem Icon und vertikale Anzeige der Icon) oder *Arrange.Horizontal* (Text neben dem Icon und horizontale Anzeige der Icon) annehmen.

Methoden einer IconView:

Methoden	Beschreibung
Add ( Key As String, Text As String [ , Picture As Picture, After As String ] ) As _IconView_Item	Fügt ein Element (Item) in die IconView ein. Key ist der Schlüssel des neuen Elements. Text ist der Text des neuen Elements. Picture und After sind optionale Parameter. Mit Picture wird das Bild über dem Text angezeigt. Ist der Parameter After gesetzt, dann erfolgt das Einfügen an der durch After definierten Position, sonst am Ende der IconView.
Clear()	Löscht den Inhalt der IconView.
Exist ( Key As String ) As Boolean	Gibt True zurück, wenn das Element mit dem angegebenen Schlüssel existiert.
Remove(Key As String)	Löscht das Element mit dem angegebenen Schlüssel aus der IconView.
SelectAll()	Selektiert und markiert alle Elemente in der IconView in Abhängigkeit von der .Mode-Eigenschaft.
Unselect()	Hebt die Auswahl der markierten Elemente in der IconView auf.
MoveFirst()	Bewegt den internen Cursor auf das erste Element der IconView. Die Funktion liefert True, wenn die IconView <i>leer</i> ist.
MoveNext()	Bewegt den internen Cursor auf das nächste Element der IconView. Die Funktion liefert True, wenn das nächste Element <i>nicht</i> existiert.
MovePrevious()	Bewegt den internen Cursor auf das vorherige Element der IconView. Die Funktion liefert True, wenn das vorherige Element <i>nicht</i> existiert.
MoveTo ( Key As String )	Bewegt den internen Cursor auf das Element mit dem angegebenen Schlüssel. Die Funktion liefert True, wenn das Element <i>nicht</i> existiert.
MoveBack()	Wenn eine von den o.a. Move-Methoden versagt (Rückgabewert True), so können Sie diese Funktion nutzen, um den internen Cursor an seine ursprüngliche Position zu setzen. Liefert True, wenn der interne Cursor <i>nicht</i> an einer gültigen Position war.

Tabelle 17.6.1.2: Übersicht zu ausgewählten Methoden der Klasse IconView

### 17.6.2 Ereignisse IconView

Spezielle Ereignisse der Komponente *IconView* finden Sie hier:

Ereignis	Beschreibung
Activate	Wird ausgelöst, wenn der Benutzer auf ein Icon in der IconView <u>doppelt</u> klickt.
Cancel	Dieses Ereignis wird ausgelöst, wenn der Benutzer eine Umbenennung abgebrochen hat.
Click	Wird ausgelöst, wenn ein Element in der IconView angeklickt und somit ausgewählt wird.
Compare	Dieses Ereignis wird ausgelöst, wenn zwei Elemente einer IconView verglichen werden. Das Ergebnis des Vergleichs muss in der Compare-Eigenschaft der IconView gespeichert werden.
Select	Wird ausgelöst, wenn sich die Selektion ändert.
Rename	Wird ausgelöst, wenn der Benutzer einen Eintrag umbenannt hat.

Tabelle 17.6.2.1: Übersicht zu Ereignissen einer IconView

### 17.6.3 Projekt 1 – IconViewCompare

Mit Hilfe der Eigenschaft *IconView.Sorted* können Sie die Elemente in einer IconView nach dem angezeigten Text sortiert anzeigen lassen. Über die Eigenschaft *IconView.Compare* und das Ereignis *IconView\_Compare(Key As String, OtherKey As String)* gelingen Ihnen auch Sortierungen nach selbst festgelegten Kriterien. Verwenden Sie die Eigenschaft *Compare*, um das Ergebnis des Vergleichs eines Compare-Ereignisses anzuzeigen. Dabei gilt für den Rückgabewert des Vergleichs:

- 0 bedeutet, dass die beiden Elemente gleich sind.
- 1 bedeutet, dass das erste Element nach dem zweiten steht.
- -1 bedeutet, dass das erste Element vor dem zweiten steht.



Abbildung 17.6.3.1: IconView mit drei Elementen

Die Reihenfolge der Anzeige beim Programmstart wird durch die Rangfolge beim Füllen der IconView vorgegeben → Abbildung 17.6.3.1.



Abbildung 17.6.3.2: IconView mit sortierten Elementen

Nach dem Sortieren mit dem Sortierkriterium  $Bild.Breite * Bild.Höhe$  hat sich die Anzeige geändert → Abbildung 17.6.3.2. Der Quelltext wird vollständig angegeben:

```
' Gambas class file

Public Sub Form_Open()
  Dim aPaths As String[] = ["icon:/48/book", "icon:/64/calendar", "icon:/32/play"]
  Dim sPath As String

  FMain.Center
  FMain.Resizable = False
  ' Als Key wird der Pfad zum Bild gewählt.
  For Each sPath In aPaths
    ivwIcons.Add(sPath, File.BaseName(sPath), Picture[sPath]) ' Füllen der IconView
  Next ' sPath
End ' Form_Open()

Public Sub ivwIcons_Compare(Key As String, OtherKey As String)
  Dim iSize, iOtherSize As Integer
  ' Sortier-Kriterium: Bild.Breite * Bild.Höhe
  With ivwIcons[Key].Picture
    iSize = .W * .H
  End With
  With ivwIcons[OtherKey].Picture
    iOtherSize = .W * .H
  End With
  ivwIcons.Compare = IIf(iSize = iOtherSize, 0, IIf(iSize > iOtherSize, 1, -1))
  ' ivwIcons.Compare = Sgn(iSize - iOtherSize) ' Alternative:
End

Public Sub btnSort_Click()
  ' Nach dem Setzen von ivwIcons.Sorted auf True wird das IconView im Hintergrund
  ' eine Reihe von Compare-Events auslösen, um die Liste sortieren zu lassen.
  ' Wird ivwIcons.Sorted auf False gesetzt, dann wird nicht weiter sortiert.

  ivwIcons.Sorted = Not ivwIcons.Sorted
  If ivwIcons.Sorted Then
    btnSort.Text = ("Icon sind sortiert.")
    btnSort.Enabled = False
  Endif ' ivwIcons.Sorted = True ?
End ' btnSort_Click()
```

17.6.4 Projekt 2 - IconViewEdit

Wenn Sie die Eigenschaft *IconView.Editable* auf True setzen, können Sie ein markiertes Element mit F2 in den Edit-Modus setzen, den Text ändern und mit Enter den Edit-Modus wieder verlassen. Der geänderte Text wird übernommen:

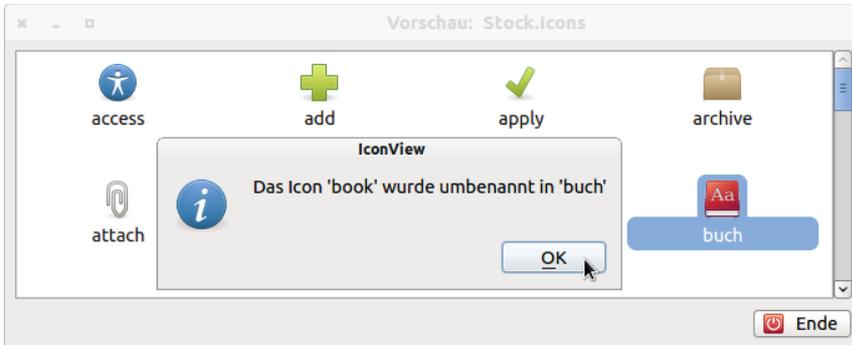


Abbildung 17.6.4.1: IconView – Elemente umbenennen

Der Auszug aus dem Quelltext ermöglicht den Edit-Modus für ein IconView mit *Speicherung* des alten Textes des umbenannten Icon:

```
Private $sLastIconName As String

Public Sub Form_Open()
    FMain.Center
    FMain.Resizable = True
    ...
    IconView.Editable = True
End ' Form_Open()

Public Sub IconView_Rename()
    Message.Info("Das Icon '" & $sLastIconName & "' wurde umbenannt in '" & IconView.Current.Text & "'")
End ' IconView_Rename()

Public Sub IconView_Select()
    $sLastIconName = IconView.Current.Text
End ' IconView_Select()
```