

### 12.6.0 Klasse TabStrip

Die Klasse *TabStrip* (*gb.qt4*) implementiert ein Register-Steurelement. Der Vorteil dieses Steuer-Elements liegt darin, mehrere Dialoge eines Programms auf einzelnen, hintereinander liegenden Register-Karten abzubilden und über die Registerkarten-Reiter eine schnelle Navigation zu realisieren.

Im nächsten Bild sehen Sie die kompakte Programm-Oberfläche für die Administration eines Servers, bei der zum Beispiel in 13 Registern über 35 Register-Karten eingesetzt wurden:

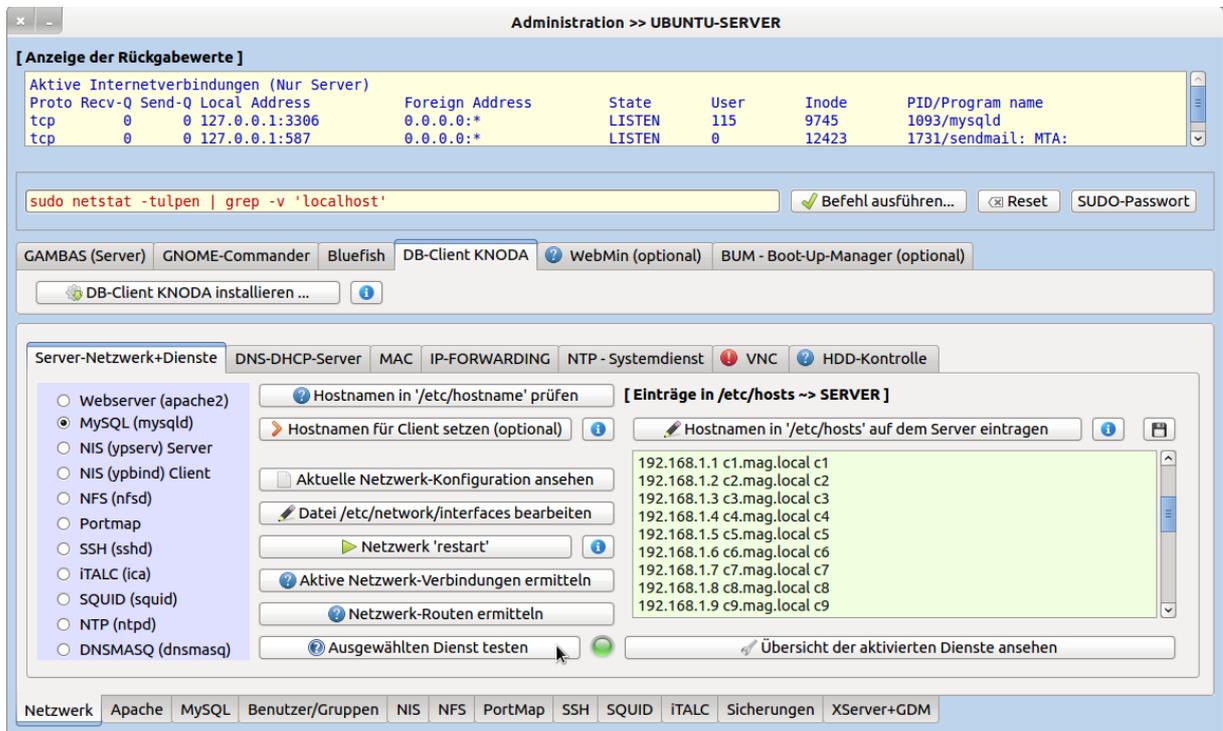


Abbildung 12.6.0.1: Programm zur Administration eines Schul-Servers

#### 12.6.0.1 Eigenschaften

Im nächsten Abschnitt werden nur wesentliche Eigenschaften der Klasse *TabStrip* (*gb.qt4*) vorgestellt:

Eigenschaft	Datentyp	Beschreibung
Text	String	Setzt den Text im Registerkarten-Reiter der <u>aktuellen</u> Register-Karte oder gibt den Text zurück.
Caption	String	Synonym für die Eigenschaft <i>Text</i> .
Picture	Picture	Setzt ein Icon im Registerkarten-Reiter der <u>aktuellen</u> Register-Karte oder gibt das Bild zurück.
Current	<i>.TabStripContainer</i>	Gibt die <u>aktuelle</u> Register-Karte zurück.
Index	Integer	Setzt den Index für die <u>aktuelle</u> Register-Karte oder gibt diesen Wert zurück.

Eigenschaft	Datentyp	Beschreibung
Count	Integer	Setzt die Anzahl der Register-Karten im Register oder gibt deren Anzahl zurück.
Children	<i>.Container.Children</i>	Gibt eine Sammlung <u>aller</u> Steuerelemente im Register zurück.
Closable	Boolean	Ist der Wert True, so erhalten <u>alle</u> Registerkarten-Reiter eine Schließen-Schaltfläche. Sie können den Wert auch auslesen.
Orientation	Integer	Setzt die Ausrichtung der Registerkarten-Reiter auf links, rechts, oben oder

Eigenschaft	Datentyp	Beschreibung
		unten oder gibt diesen Wert zurück. Standard ist 'oben' – voreingestellt über die Konstante 'Align.Top'. Einsetzbar sind diese vier Konstanten: Align.Top, Align.Bottom, Align.Left oder Align.Right.

Tabelle 12.6.0.1.1 : Eigenschaften der Klasse TabStrip

Beachten Sie, dass sich die Eigenschaften im oberen Teil der Tabelle stets auf die *aktuelle* Register-Karte beziehen!

### 12.6.0.2 Methoden

Von den Methoden der Klasse TabStrip ist nur diese interessant:

```
TabStrip.FindIndex (gb.qt4)
Function FindIndex (Child As Control) As Integer
```

Mit dieser Methode ermitteln Sie den Index des angegebenen Steuerelements in einem Register. Wird das Steuerelement nicht gefunden, so wird als Funktionswert -1 zurückgegeben.

### 12.6.0.3 Ereignisse

Das Ereignis *Close(Index As Integer)* der Klasse TabStrip wird nur dann ausgelöst, wenn eine bestimmte Register-Karte geschlossen wird. Das setzt voraus, dass die Eigenschaft *TabStrip.Closable* auf den Wert 'True' gesetzt wurde (Standard ist 'False'). Als Argument wird der Index der zu schließenden Register-Karte übergeben.

### 12.6.0.4 .TabStripContainer

Die virtuelle Klasse *.TabStripContainer* repräsentiert eine Register-Karte in einer TabStrip-Komponente (Register). Sie können einen *TabStrip* wie ein Array benutzen, um so ein Objekt der Klasse *.TabStripContainer* (also eine Register-Karte) über seinen Index zu erhalten. Die Klasse *.TabStripContainer* hat sechs Eigenschaften und verfügt nur über eine Methode.

Eigenschaft	Datentyp	Beschreibung
Text	String	Setzt den Text im Register-Reiter der Register-Karte oder gibt den Text zurück.
Caption	String	Synonym für die Eigenschaft <i>Text</i> .
Children	<i>.TabStripContainer.Children</i>	Sammlung aller Komponenten auf der Register-Karte.
Enabled	Boolean	Gibt mit dem Wert 'True' an, ob die Register-Karte aktiv ist.
Visible	Boolean	Gibt mit dem Wert 'True' an, ob die Register-Karte sichtbar ist.
Picture	Picture	Setzt ein Icon im Register-Reiter der Register-Karte oder gibt das Bild zurück.

Tabelle 12.6.0.4.1 : Eigenschaften der virtuellen Klasse .TabStripContainer

Die Methode *Delete* löscht die aktuelle Register-Karte.

### 12.6.0.5 .TabStripContainer.Children

Die virtuelle Klasse *.TabStripContainer.Children* repräsentiert eine Sammlung aller Komponenten auf einer Register-Karte. Diese Klasse besitzt nur eine Eigenschaft: *Count*, welche die Anzahl der Komponenten auf einer Register-Karte zurück gibt. Bedenken Sie jedoch, dass beispielsweise bei einer Container-Komponente nur diese Container-Komponente mitgezählt wird, aber keine der Komponenten im Container!

Sie können über alle Komponenten auf den Register-Karten eines Registers iterieren und sich zum Beispiel die Namen der Komponenten ausgeben lassen. Im Projekt1 des Kapitels 12.6.1 wird genau das realisiert und liefert dieses Ergebnis → Abbildung 12.6.0.5.2, wenn zum Beispiel in einem Register eine Register-Karte *nicht* sichtbar ist:

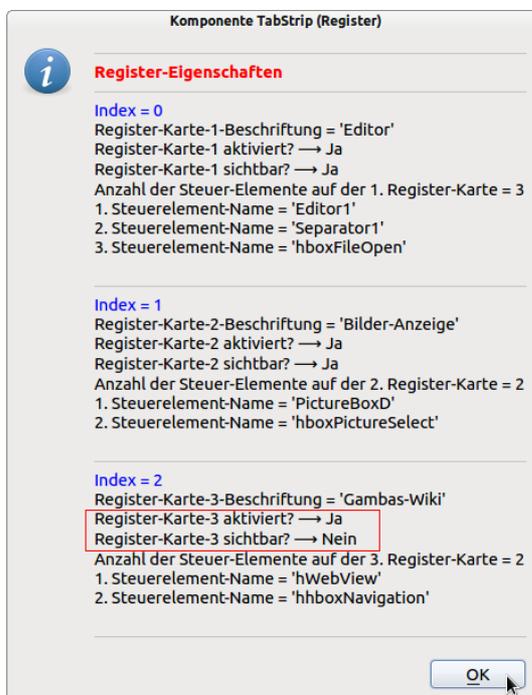
Abbildung 12.6.0.5.1: Register-Karte 3 existiert – wird aber *nicht* angezeigt

Abbildung 12.6.0.5.2: Anzeige ausgewählter Register-Eigenschaften

Mit diesem Quelltext-Ausschnitt können Sie ausgewählte Register-Eigenschaften in einer Message-Box oder in der Konsole der Gambas-IDE anzeigen:

```
Public Sub btnGetInformations_Click()
    Dim i, k As Integer
    Dim hControl As Control
    Dim sMessage As String

    sMessage = "<hr><b><font color='Red'>Register-Eigenschaften</font></b><hr>"
    For i = 0 To TabStrip1.Count - 1
        sMessage &= "<font color='Blue'>Index = " & CStr(i) & "</font>"
        sMessage &= "<br>Register-Karte-" & CStr(i + 1) & "-Beschriftung = '"
        sMessage &= TabStrip1[i].Caption & "'"
        sMessage &= "<br>Register-Karte-" & CStr(i + 1) & " aktiviert? " & String.Chr(10230)
        sMessage &= IIf(TabStrip1[i].Enabled = "T", " Ja", " Nein")
        sMessage &= "<br>Register-Karte-" & CStr(i + 1) & " sichtbar? " & String.Chr(10230)
        sMessage &= IIf(TabStrip1[i].Visible = "T", " Ja", " Nein")
        sMessage &= "<br>Anzahl der Steuer-Elemente auf der " & CStr(i + 1)
        sMessage &= ". Register-Karte = " & CStr(TabStrip1[i].Children.Count)
        k = 1
        For Each hControl In TabStrip1[i].Children
```

```

        sMessage &= "<br>" & CStr(k) & ". Steuerelement-Name = '" & hControl.Name & "' "
        Inc k
    Next
    sMessage &= "<br><hr>"
Next
Message.Info(sMessage)

' Anzeige der Informationen in der Konsole in der IDE
' For i = 0 To TabStrip1.Count - 1
'     Print "Index = "; i
'     Print "Register-Karte-"; i + 1; "-Beschriftung = "; ""; TabStrip1[i].Caption; ""
'     Print "Register-Karte-"; i + 1; " aktiviert? --> "; TabStrip1[i].Enabled
'     Print "Register-Karte-"; i + 1; " sichtbar? --> "; TabStrip1[i].Visible
'     Print "Anzahl der Steuer-Elemente auf der "; i + 1; ". Register-Karte = "; TabStrip1[i].Children.Count
'     k = 1
'     For Each hControl In TabStrip1[i].Children
'         Print k; ". Steuerelement-Name = "; ""; hControl.Name; ""
'         Inc k
'     Next
'     Print
' Next
End ' btnGetInformations_Click()

```

Beachten Sie: Wenn Sie zur Laufzeit mit der Maus über einem Registerkarten-Reiter stehen, so können Sie mit dem Mausekranz durch die einzelnen Register-Karten navigieren.