

18.14 StatusBar

Die Klasse *StatusBar* von Raymond de Bruijne (<http://gambas-club.de/viewtopic.php?f=4&t=2928>) setzt auf ein Panel als Container für eine StatusBar am unteren Fensterrand (Standardeinstellung), bei der Label zur Anzeige von Text eingefügt werden können. Jedes Label kann in besonderer Weise – in Abhängigkeit vom Anzeigezweck – formatiert werden:

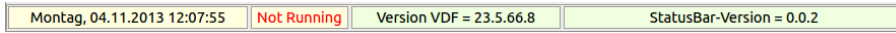


Abbildung 18.14.1: Container StatusBar mit 4 Labeln

Das Besondere der StatusBar liegt darin, dass die einzelnen Label anklickbar sind und damit weitere Informationen – zum Beispiel in einer MessageBox – angezeigt werden können:

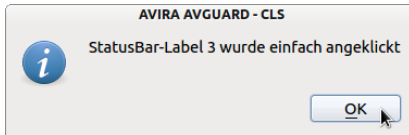


Abbildung 18.14.2: Information zum Label 3

Außerdem können Sie bei jedem Label in der StatusBar (→ Abbildung 18.14.1)

- einen Text (optional) anzeigen lassen,
- eine eigene Schreib- und Hintergrundfarbe für den Text angeben,
- die Ausrichtung des (optionalen) Textes vorgeben und
- die Randeigenschaften zuweisen.

18.14.1 Eigenschaften StatusBar

Die Klasse *StatusBar* besitzt nur die eine Eigenschaft *Version* (Datentyp String, readonly), mit der Sie die aktuelle Version der Klasse abrufen können.

18.14.2 Methoden StatusBar

Die Methoden der Klasse *StatusBar* erschließen sich am einfachsten über den Quelltext der Klasse:

```
[1] ' Gambas class file
[2]
[3] '*****
[4] ' Klassenname: StatusBar.class
[5] ' Datum:      18.03.2011 - 4.11.2013
[6] ' Autor:      Raymond de Bruijne
[7] ' Anpassung:  Dr. Hans-Joachim Lehmann
[8] ' EMail:      rdbruijne@freenet.de, hans@gambas-buch.de
[9] '*****
[10] ' Eigenschaften
[11] '*****
[12] Property Read Version As String
[13]
[14] '*****
[15] ' (Arbeits-)Variablen
[16] '*****
[17] Private $Version As String = "0.1.2"           ' Versionsinformation
[18] Private $StatusBarHeight As Integer = 28       ' StatusBarhöhe
[19] Private $MenuBarHeight As Integer = 24        ' Menübarhöhe
[20] Private $StatusBarContainer As Panel          ' Panel als StatusBar-Container
[21] Private $Count As Integer                    ' Zähler für Labels
[22] Private $Label[255] As Label                 ' Maximal 255 Labels verfügbar
[23] Private $LabelWidth[255] As Integer          ' Speicher Label-Breite
[24] Private $LabelSpace As Integer = 3           ' Abstand zwischen den einzelne Labels
[25] Private $Offset As Integer = 2               ' Offset-Wert für Rahmen
[26]
[27] '*****
[28] ' Methoden
[29] '*****
[30] Private Function Version_Read() As String
[31]     Return $Version
[32] End ' Version_Read()
[33]
[34] Public Sub _new(Parent As Container, PanelWidth As Integer, Optional Text As String)
[35]
```

```

[36] ' -----
[37] ' StatusBar-Container erzeugen
[38] ' -----
[39] $StatusBarContainer = New Panel(Parent)
[40] $StatusBarContainer.Border = Border.Sunken
[41]
[42] ' -----
[43] ' 1. Label erzeugen
[44] ' -----
[45] Inc $Count
[46] $Label[1] = New Label($StatusBarContainer)
[47] StatusBarInit($Count, Parent)
[48] With $Label[1]
[49]     .Width = Parent.Width
[50]     .Border = Border.Sunken
[51]     .Background = Color.Background
[52]     .Width = PanelWidth
[53]     .Text = Text
[54] End With
[55] $LabelWidth[1] = PanelWidth
[56] BuildStatusBar(Parent)
[57]
[58] End ' _new(..)
[59]
[60] Private Sub BuildStatusBar(hParent As Container)
[61]
[62]     Dim $Counter As Integer
[63]     Dim $PositionX As Integer = $OffSet
[64]     Dim $RestFensterbreite As Integer
[65]
[66]     ' -----
[67]     ' StatusBar
[68]     ' -----
[69]     If hParent.Window.Menus.Count > 0 Then
[70]         ' Falls im Elternfenster ein Menüleiste vorhanden ist:
[71]         $StatusBarContainer.Move(0, hParent.H - ($StatusBarHeight + $MenubarHeight - $OffSet),
[72]             hParent.Width, 26)
[73]     Else
[74]         ' Das Gleiche ohne Menüleiste:
[75]         $StatusBarContainer.Move(0, hParent.H - $StatusBarHeight - $OffSet, hParent.Width, 26)
[76]     Endif ' hParent.Window.Menus.Count > 0 ?
[77]
[78]     ' -----
[79]     ' Label einfügen
[80]     ' -----
[81]     If $Count = 1 Then
[82]         ' Bei nur einem Label die gesamte (verfügbare) Fensterbreite ausnutzen
[83]         $Label[1].Move($OffSet, $OffSet, hParent.Width - (2 * $OffSet), 22)
[84]     Else
[85]         ' Labels einfügen ...
[86]         For $Counter = 1 To $Count
[87]             If $Counter = 1 Then
[88]                 ' Das erste Label an den Anfang setzen...
[89]                 $Label[$Counter].Move($OffSet, $OffSet, $LabelWidth[$Counter] - (2 * $OffSet), 22)
[90]                 $PositionX += $Label[$Counter].Width + $LabelSpace
[91]             Else
[92]                 ' ...und anschliessend weitere Labels einfügen
[93]                 $Label[$Counter].Move($PositionX, $OffSet, $LabelWidth[$Counter] - (2 * $OffSet), 22)
[94]                 $PositionX += $Label[$Counter].Width + $LabelSpace
[95]             Endif ' $Counter = 1 ?
[96]             ' ...und zum Schluß bei mehr als einem Label das letzte auf Rest-Fensterbreite anpassen
[97]             If $Count = $Counter Then
[98]                 $RestFensterbreite = hParent.Width - $Label[$Counter - 1].X - $Label[$Counter - 1].Width -
[99]                     (2 * $OffSet)
[100]                 $Label[$Counter].Width = $RestFensterbreite - $OffSet
[101]             Endif ' $Count = $Counter ?
[102]         Next ' $Counter
[103]     Endif ' $Count = 1
[104]
[105] End ' BuildStatusBar(..)
[106]
[107] Public Sub Add(hParent As Container, Width As Integer, Optional Text As String)
[108]
[109]     Inc $Count
[110]     $Label[$Count] = New Label($StatusBarContainer)
[111]     With $Label[$Count]
[112]         .Width = Width
[113]         .Text = Text
[114]         .Border = Border.Sunken
[115]         .Background = Color.Background
[116]     End With
[117]     $LabelWidth[$Count] = Width
[118]     StatusBarInit($Count, hParent)
[119]     BuildStatusBar(hParent)

```

```
[118]
[119] End ' Add(..)
[120]
[121] Private Sub StatusBarInit(Id As Integer, Parent As Container)
[122]     Object.Attach($Label[Id], Parent, "StatusLabel" & Id) ' Event-Name ist 'StatusLabel'
[123] End ' StatusBarInit(..)
[124]
[125] Public Sub Resize(hParent As Container)
[126]     BuildStatusBar(hParent)
[127] End ' Resize(..)
[128]
[129] Public Sub SetText(Id As Integer, Text As String)
[130]     $Label[Id].Text = Text
[131] End ' SetText(..)
[132]
[133] Public Sub SetBorder(Id As Integer, Border As Integer)
[134]     $Label[Id].Border = Border
[135] End ' SetBorder(..)
[136]
[137] Public Sub SetAlignment(Id As Integer, Alignment As Integer)
[138]     $Label[Id].Alignment = Alignment
[139] End ' SetAlignment(..)
[140]
[141] Public Sub SetToolTip(Id As Integer, Text As String)
[142]     $Label[Id].ToolTip = Text
[143] End ' SetToolTip(..)
[144]
[145] Public Sub SetBackground(hParent As Container, Label_Id As Integer, Value As Integer)
[146]     $Label[Label_Id].Background = Value
[147]     BuildStatusBar(hParent)
[148] End ' SetBackground(..)
[149]
[150] Public Sub SetForeground(hParent As Container, Label_Id As Integer, Value As Integer)
[151]     $Label[Label_Id].Foreground = Value
[152]     BuildStatusBar(hParent)
[153] End ' SetForeground(..)
```

18.14.3 Ereignisse StatusBar

Sie können für jedes Label nach folgender Syntax ein *spezifisches* Ereignis generieren, wobei die Ereignisse denen eines Labels entsprechen. Für alle ID's ist der Event-Name StatusLabel:

```
Public Sub Event-Name&Label-ID&_Ereignis
... Ereignisbehandlungsroutine
End
```

Hier die konkrete Umsetzung für den Klick mit der linken Maustaste auf das 3. Label, da für ein Label kein Click()-Ereignis existiert:

```
' Ereignis auslösen - Label 3 - Einfachklick LMT
Public Sub StatusLabel3_MouseDown()
    If Mouse.Left Then Message.Info("StatusBar-Label 3 wurde einfach angeklickt")
End ' StatusLabel3_MouseDown()
```

Was Sie in der Mitteilungsbox anzeigen lassen, hängt vor allem vom Programm-Kontext ab.

18.14.4 Projekt CLS VirenScanner

Das Projekt 'CLS VirenScanner' stellt ein GUI für den Avira Kommandozeilenscanner zur Verfügung. Der Quelltext wird nur in den Auszügen angegeben, die für das Verständnis des Einsatzes der Klasse *StatusBar* nötig sind:

```
' Gambas class file

Private myStatusBar As StatusBar

Public Sub Form_Open()

    FMain.Center
    Timer1.Delay = 1000
    Timer1.Start

    myStatusBar = New StatusBar(Me, 240, Format$(Now, "dddd, dd/mm/yyyy hh:nn:ss"))
    myStatusBar.SetBackground(Me, 1, &HFFFFFF)
    myStatusBar.SetForeground(Me, 1, &H000000)
    myStatusBar.SetToolTip(1, "Datum&Zeit")
    myStatusBar.SetAlignment(1, Align.Center)
```

```

myStatusBar.Add(Me, 100, "Not Running")
myStatusBar.SetBackground(Me, 2, &HFFFFFF)
myStatusBar.SetForeground(Me, 2, Color.Red)
myStatusBar.SetToolTip(2, "Programm-Status")
myStatusBar.SetAlignment(2, Align.Center)

myStatusBar.Add(Me, 210, "Version VDF = 23.5.66.8")
myStatusBar.SetBackground(Me, 3, &HFFFFFF)
myStatusBar.SetAlignment(3, Align.Center)

' Die Feldlänge wird beim letzten Feld automatisch gesetzt -> 0 eintragen!
myStatusBar.Add(Me, 0, "StatusBar-Version = " & myStatusBar.Version)
myStatusBar.SetBackground(Me, 4, &HFFFFFF)
myStatusBar.SetBorder(4, Border.Plain)
myStatusBar.SetAlignment(4, Align.Center)

FileView1.ShowDetailed = True
lblDirectory.Text = DirView1.Current

End ' Form_Open()

'*****

' Text setzen - Label 1
Public Sub Timer1_Timer()
    myStatusBar.SetText(1, Format$(Now, "dddd, dd/mm/yyyy hh:nn:ss"))
End ' Timer1_Timer() : Aktuelle Zeit im Panel 1 anzeigen

' Methode aufrufen
Public Sub Form_Resize()
    myStatusBar.Resize(Me)
End ' Form_Resize()

' Ereignis auslösen - Label 1 - Doppelklick
Public Sub StatusLabel1_DblClick()
    Message("StatusBar-Label 1 wurde doppelt angeklickt!")
End ' StatusPanel1_DblClick()

' Ereignis auslösen - Label 3 - Einfachklick LMT
Public Sub StatusLabel3_MouseDown()
    If Mouse.Left Then Message.Info("StatusBar-Label 3 wurde einfach angeklickt")
End ' StatusPanel3_MouseDown()

'*****
'... Quelltext GUI Scanner - Ausschnitt

Public Sub DirView1_Select()
    FileView1.Dir = DirView1.Current
    lblDirectory.Text = DirView1.Current
    lblFile.Text = ""
End ' DirView1_Select()

Public Sub FileView1_Click()
    lblFile.Text = DirView1.Current & / FileView1.Current
End ' FileView1_Click()

Public Sub btnClose_Click()
    FMain.Close
End ' btnClose_Click()

```