

20.15 Klasse Desktop (gb.qt4)

Die Klasse *Desktop* (*gb.qt4*) besitzt nur wenige Eigenschaften sowie eine Methode und eine Konstante. Diese können Sie nutzen, um Informationen zum Bildschirm und zum Desktop zu ermitteln.

Diese Klasse ist immer im Zusammenhang mit der Klasse *Desktop* der Komponente Desktop sowie den Klassen *Screens* und *Screen* zu betrachten.

20.15.1 Eigenschaften

Alle Eigenschaften sowie deren Beschreibung finden Sie in der folgenden Tabelle:

Eigenschaft	Type	Beschreibung
.Height	Integer	Gibt die Höhe des nutzbaren Desktop auf einem Monitor an. Das nutzbare Desktop-Rechteck ist die Bildschirmfläche vermindert um alle stationären Desktop-Panels. Äquivalent zu <i>Screens[0].AvailableHeight</i>
.Width	Integer	Gibt die Breite des nutzbaren Desktop auf einem Monitor an. Das nutzbare Desktop-Rechteck ist die Bildschirmfläche vermindert um alle stationären Desktop-Panels. Äquivalent zu <i>Screens[0].AvailableWidth</i>
.Resolution	Integer	Gibt die Auflösung des Bildschirms in DPI (Dots Per Inch) an.
.Scale	Integer	Gibt die halbe Höhe des Standard-Desktop-Fonts in Pixeln zurück.
.HasSystemTray	Boolean	Gibt True zurück, wenn der Desktop eine TaskLeiste hat.

Tabelle 20.15.1.1: Übersicht zu den Eigenschaften der Klasse Desktop (gb.qt4)

Dem Vorschlag in der Gambas-Dokumentation folgend können Sie mit der folgenden Prozedur die Eigenschaften des Desktops auslesen und in einer MessageBox anzeigen lassen:

```
Public Sub btnGetDesktopInformation_Click()
  Dim sMessage, sLabel As String

  sMessage = "<hr><b>Desktop-Eigenschaften</b><hr>"
  sMessage &= "Nutzbare Desktop-Höhe = " & Desktop.Height & " Pixel"
  sMessage &= "<br>Nutzbare Desktop-Breite = " & Desktop.Width & " Pixel"
  sMessage &= "<br>Desktop-Top = " & Desktop.Y
  sMessage &= "<br>Desktop-Left = " & Desktop.X & "<hr>"
  sMessage &= "Zeichensatz = " & Desktop.Charset

  sLabel = IIf(Desktop.HasSystemTray = False, " Nein", " Ja.")
  sMessage &= "<br>SystemTray = " & sLabel & " Desktop.HasSystemTray

  sMessage &= "<br>Skalierungsfaktor = " & Desktop.Scale
  sMessage &= "<br>Auflösung = " & Desktop.Resolution & " DPI <hr>"
  Message.Info(sMessage)
End ' btnGetDesktopInformation_Click()
```



Abbildung 20.15.1.2: Anzeige Eigenschaften Desktop

20.15.2 Konstante Desktop.Charset

Die Konstante *Desktop.Charset* gibt den Zeichensatz an, der von der grafischen Oberfläche zum Anzeigen von Text verwendet wird. Die QT- und GTK+-Komponenten verwenden den UTF-8-Zeichen-

satz. Sie müssen vorsichtig sein, weil das zugrunde liegende Betriebssystem einen anderen Zeichensatz verwenden könnte!

20.15.3 Methode Desktop.ScreenShot (Bildschirm-Kopie)

Die angegebene Methode gibt nicht nur eine Bildschirm-Kopie des vollständigen Desktop als Objekt vom Typ *Picture* zurück, sondern ermöglicht als Option auch die Kopie eines ausgewählten Teiles davon:

```
Static Function Screenshot ( [ X As Integer, Y As Integer, Width As Integer, Height As Integer ] ) As Picture
```

Diese Prozedur gibt eine Bildschirm-Kopie zurück, die sofort in einer *PictureBox* angezeigt wird und dann in einem bestimmten Verzeichnis die aktuelle Bildschirmkopie als png-Grafik mit höchster Qualität (→ 100) abspeichert.

```
[1] Public Sub btnGetScreenShot_Click()
[2]     FMain.Hide
[3]     Wait 0.1 ' In der Praxis bis auf 3 Sekunden erhöhen!
[4]     PictureBox1.Picture = Desktop.Screenshot()
[5]     PictureBox1.Picture.Save(User.Home & "current_screenshot.png", 100)
[6]     FMain.Show
[7]
[8] ' Alternative ohne direkte Anzeige - Option Speichern-Dialog
[9] ' FMain.Hide
[10] ' Wait 0.05
[11] ' Desktop.Screenshot().Save(User.Home & "current_screenshot.png", 100)
[12] ' FMain.Show
[13]
[14] End ' btnGetScreenShot_Click()
```

Mit einem noch hinzuzufügenden Speichern-Dialog können Sie mehrere Bildschirm-Kopien komfortabel abspeichern.

Wenn Sie auf die Anzeige verzichten, dann gelingt es über die Speichern-Prozedur *Desktop.Screenshot.Save(Path, Quality)* die Bildschirm-Kopie sofort im angegebenen Bild-Pfad und in der geforderten Qualität zu speichern.

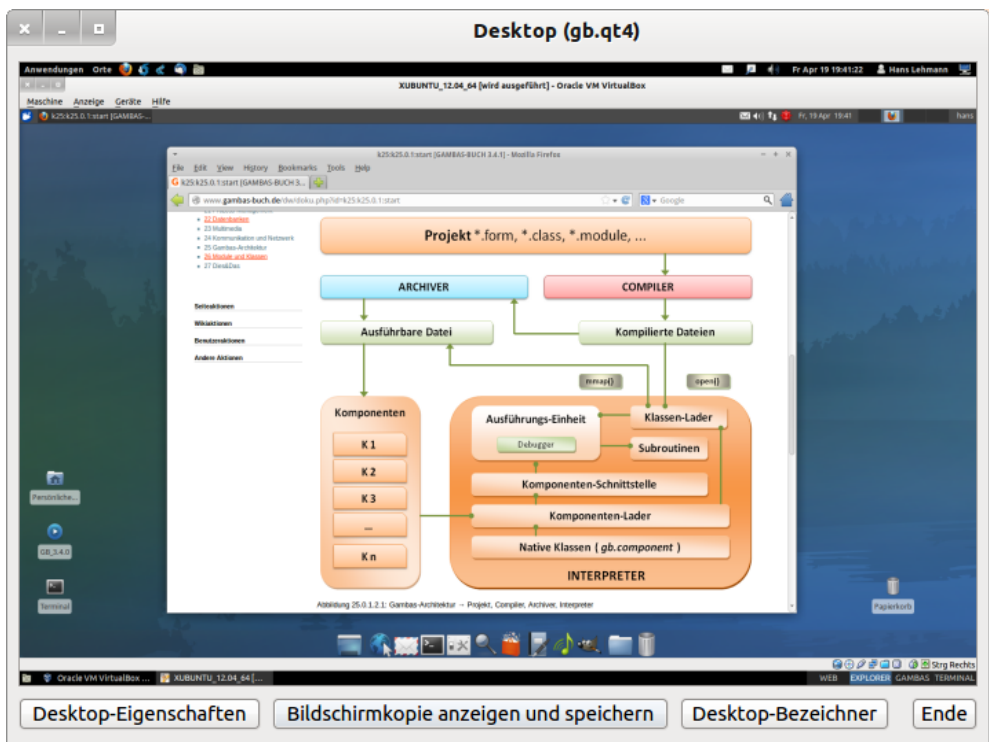


Abbildung 20.15.3.1: Desktop-Projekt: Anzeige Bildschirm-Kopie (Xubuntu in einer VM) in einer PictureBox

20.15.4 Erweiterungen

Sie können auch die Namen der virtuellen Desktops ermitteln, wenn Sie das Programm *wmctrl* auf Ihrem System installiert haben. Mit dem Befehl:

```
hans@linux:~$ wmctrl -d
```

erhalten Sie mit der Option *-d* folgende Ausgabe, bei der der aktuelle Desktop mit einem **Stern** gekennzeichnet ist und jeweils am Zeilenende der Name des Desktops steht:

```
0 - DG: 1680x1050 VP: N/A WA: 0,24 1680x1002 WEB
1 - DG: 1680x1050 VP: N/A WA: 0,24 1680x1002 EXPLORER
2 * DG: 1680x1050 VP: 0,0 WA: 0,24 1680x1002 GAMBAS
3 - DG: 1680x1050 VP: N/A WA: 0,24 1680x1002 TERMINAL
```

Für eine Gambas-Prozedur – die das Gleiche leistet – setzen Sie vorteilhaft die Instruktion **Shell** oder **Exec** (→ Kapitel 21.3.1 Einsatz Quick-Syntax) sowie einige Zeichenketten und Arrays ein, um aus der zurückgelieferten Zeichenkette – gespeichert in der Variablen *sListe* – die Desktop-Bezeichner zu ermitteln und anzuzeigen:

```
Public Sub btnGetDesktopName_Click()
    Dim sElement, sListe, sMessage As String
    Dim aMatrix, aZeile As New String[]

    Shell "wmctrl -d" To sListe

    If sListe Then aMatrix = Split(sListe, "\n")
    sMessage = "<hr><b>Desktop-Bezeichner</b><hr>"

    For Each sElement In aMatrix
        If sElement Then
            aZeile = Split(sElement, " ")
            sMessage &= "<br>" & aZeile[aZeile.Count - 1]
        Endif
    Next ' sElement

    Message.Info(sMessage)
End ' btnGetDesktopName_Click()
```

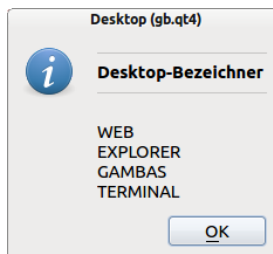


Abbildung 20.15.4.1: Anzeige der Desktop-Namen (Bezeichner)