

19.7.1 Klasse Desktop (gb.desktop)

Die Klasse *Desktop* der gleichnamigen Komponente erlaubt Ihnen auf der Basis der xdg-utils-Skripte des Portland-Projekts die Verwendung von auf den Desktop bezogenen Routinen. Die Anzahl der Eigenschaften und Methoden ist überschaubar. Ausgewählte Eigenschaften und Methoden werden jeweils in einer Tabelle aufgeführt und beschrieben:

Eigenschaft	Typ	Beschreibung
.ActiveWindow	Integer	Das aktive aktuelle Top-Level-Fenster wird zurückgegeben oder gesetzt.
.Count	Integer	Die Anzahl virtueller Desktops wird zurückgegeben oder gesetzt.
.Current	Integer	Die Nummer des aktuellen Desktops wird zurückgegeben oder gesetzt.
.Passwords	_DesktopPasswords	Gibt ein virtuelles Objekt zurück. <i>Gegenwärtig sind weder Eigenschaften (außer Key) noch Methoden bekannt.</i>
.Path	String	Gibt den Pfad zum Desktop-Verzeichnis zurück
.RootWindow	Integer	Gibt den X11-Handle des Root-Fensters zurück
.ScreenSaver	_Desktop_ScreenSaver	Managt den Bildschirmschoner
.Showing	Boolean	Bei TRUE werden alle Fenster minimiert und Sie können den Desktop sehen
.Type	String	Gibt den Namen des aktuellen Desktops zurück (KDE, KDE4, GNOME, XFCE oder ? für einen unbekanntem Desktop)
.Windows	_Desktop_Windows	Gibt ein virtuelles Objekt zurück, das eine Liste aller Top-Level-Fenster repräsentiert

Tabelle 19.7.1.1: Übersicht zu Eigenschaften der Klasse Desktop (gb.desktop)

Die virtuelle Klasse *_Desktop_ScreenSaver* erlaubt Ihnen die Manipulation des Bildschirmschoners und hat die folgende Eigenschaft:

- *_Desktop_ScreenSaver.Enabled*
Gibt TRUE zurück, wenn der Bildschirmschoner eingeschaltet ist.

Es folgt eine Aufzählung der Methoden der virtuellen Klasse *_Desktop_ScreenSaver* und eine kurze Erläuterung der ausgelösten Aktion:

- *_Desktop_ScreenSaver.Activate* – Aktiviert den Bildschirmschoner sofort
- *_Desktop_ScreenSaver.Reset* – Deaktiviert den Bildschirmschoner sofort
- *_Desktop_ScreenSaver.Lock* – Sperrt den Bildschirmschoner sofort (→ Freischaltung Passwort)
- *_Desktop_ScreenSaver.Resume* – Bildschirmschoner-Funktion wird aktiviert
- *_Desktop_ScreenSaver.Suspend* – Bildschirmschoner-Funktion wird de-aktiviert

Eine kurze Beschreibung der Methoden der Klasse *Desktop* finden Sie in der folgenden Tabelle, der u.a. eine ausführliche Darstellung ausgewählter Methoden in weiteren (Unter-)Kapiteln folgt:

N	Methode	Beschreibung
1	.FindWindow ([Title As String, Application As String, Role As String]) As Integer[]	Es werden alle X11-Top-Level-Fenster in einem Array zurückgegeben.
2	.GetFileIcon (Path As String, Size As Integer [, Preview As Boolean]) As Picture	Für Preview gleich False wird ein Icon vom Typ Picture von der spezifizierten Datei zurückgegeben.
3	.Open (URL As String [, Wait As Boolean])	Öffnet eine Datei oder eine URL in der zugewie-

N	Methode	Beschreibung
		senen Standard-Anwendung
4	.OpenTerminal ([Dir As String])	Öffnet eine Terminal in der aktuellen Desktop-Umgebung. Es kann optional ein Start-Verzeichnis angegeben werden.
5	.RunAsRoot (Command As String)	Führt den angegebenen Befehl aus. Es wird das Root-Passwort in einem mit dem Desktop korrespondierenden Terminal angefordert.
6	.SendKeys (Keys As String)	Es wird ein Tastendruck oder eine Kombination von Tasten generiert. Keys ist eine Liste von nacheinander zu generierenden Tasten.
7	.SendMail (To As String[] [, Cc As String[], Bcc As String[], Subject As String, Body As String, Attachment As String])	Es wird eine EMail aufbereitet mit einer Empfänger-Liste, weiteren Empfängerlisten, dem Betreff, dem Text-Inhalt und einem Anhang.

Tabelle 19.7.1.2: Übersicht zu Methoden der Klasse Desktop (gb.desktop)

(1)
Die Methode FindWindow(..) gibt als Funktionswert ein Array von X11-Fenster-Kennungen zurück. Für die optionalen Parameter in der Methode gilt:

```
Desktop.FindWindow ( [ Title As String, Application As String, Role As String ] ) As Integer[]
```

- *Title* ist der Fenster-Titel. Diesen finden Sie in der X11-Fenster-Eigenschaft WM_NAME.
- *Application* ist die Fenster-Klasse. Es ist in der Regel der Name der Anwendung, die das Fenster erstellt. Er ist in der X11-Fenster-Eigenschaft WM_CLASS gespeichert.
- *Role* ist die Rolle des Fensters. Sie ist in der X11-Fenster-Eigenschaft WM_WINDOW_ROLE gespeichert.

Die Argumente *Title*, *Application* und *Role* sind reguläre Ausdrücke, wie sie im Zusammenhang mit dem LIKE-Operator (→ Kapitel 19.6.3 Exkurs LIKE und Kapitel 8.6 Spezielle Operatoren) definiert sind.

```
[1] Public Sub GetDTWindowsList()
[2]     Dim iCount As Integer
[3]     Dim DTWindow As DesktopWindow
[4]     Dim aDTWindowsList As Integer[]
[5]
[6]     txaWindowList.Clear
[7]     aDTWindowsList = Desktop.FindWindow(txbPatternBox.Text)
[8]     lblWindowCount.Text = aDTWindowsList.Count
[9]     For iCount = 0 To aDTWindowsList.Max
[10]         DTWindow = New DesktopWindow(aDTWindowsList[iCount])
[11]         If DTWindow.Id = aDTWindowsList[iCount] Then
[12]             txaWindowList.Text &= Str(iCount + 1) & "\t" & (DTWindow.Desktop + 1) &
[13]                 " " & DTWindow.Id & "\t" & DTWindow.Name & gb.NewLine
[14]         Endif ' DTWindow.Id = aDTWindowsList[iCount] ?
[15]     Next ' iCount
[16] End ' GetDTWindowsList()
[17]
[18] Public Sub btnGetDTWList_Click()
[19]     GetDTWindowsList()
[20] End ' btnGetDTWList_Click()
```

Folgende Ergebnisse konnten in einer TextArea angezeigt werden. Zuerst wurde als Suchmuster der reguläre Ausdruck `[DGM]*` (→ Zeile 7) verwendet (Fenstername beginnt mit D, G oder M):

Nummer	Desktop	Desktop.ID	Fenster-Name
1	3	52428816	DesktopWatcher 0.0.6 - Gambas 3
2	1	62914563	GHex

3	4	65011850	Mozilla Firefox
4	3	37748751	DesktopWatcher

Anschließend wurde das Suchmuster auf * festgesetzt und ergab folgende, erweiterte Liste von X11-Top-Level-Fenstern:

Nummer	Desktop	Desktop.ID	Fenster-Name
1	0	23068676	Schreibtisch
2	3	52428816	DesktopWatcher 0.0.6 - Gambas 3
3	1	62914563	GHex
4	1	60817454	Unbenanntes Dokument 1 - gedit
5	2	48234565	19.7.1.odt - LibreOffice Writer
6	4	65011850	Mozilla Firefox
7	3	37748751	DesktopWatcher
8	0	14680084	Unteres Kanten-Panel, ausgedehnt
9	0	14680067	Oberes Kanten-Panel, ausgedehnt

(2)

Die Methode GetFileIcon(..) gibt das Icon zurück, das der ausgewählten Datei zugeordnet wurde:

```
Desktop.GetFileIcon (Path As String, Size As Integer [ , Preview As Boolean ] ) As Picture
```

- *Path* ist der Datei-Pfad.
- *Size* ist die Größe des Icons (Pixel).
- *Preview* – Der Standardwert des optionalen Arguments ist FALSE und es wird die Icon-Datei vom Typ Picture zurückgegeben.

```
' Gambas class file
Public picFileIcon As Picture

Public Sub Form_Open()

    picFileIcon = Desktop.GetFileIcon(User.Home & "DesktopWatcher.gambas", 32, False)
' Speicherung des Icons in der globalen Variablen picFileIcon vom Typ Picture
    PictureBox1.W = 72
    PictureBox1.H = PictureBox1.W
    PictureBox1.Stretch = True
    PictureBox1.Picture = Desktop.GetFileIcon(User.Home & "Arbeitsfläche/GetFileIcon/
    GetFileIcon.gambas", 32, False)

End ' Form_Open
```

(5)

Mit der Methode RunAsRoot(..) verbanden viele Gambas-Programmierer die Hoffnung, nun recht einfach aus einem Gambas-Programm heraus ein Kommando ausführen zu können, das Root-Rechte erfordert. Die Umsetzung in ein Projekt ist einfach:

```
Public Sub RunAsRoot(sCommand As String)
    Desktop.RunAsRoot(sCommand)
End ' RunAsRoot(sCommand As String)

Public Sub btnRunAsRoot_Click()
    RunAsRoot("apt-get install bluefish")
End ' btnRunAsRoot_Click()
```

Es öffnet sich ein Terminal-Fenster und man wird zur (verdeckten) Eingabe des Root-Passwortes aufgefordert:

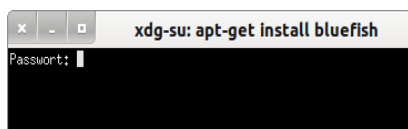


Abbildung 19.7.1.1: Terminal

Aber gegenwärtig (→ 13.4.2013) passiert nach der Eingabe des Root-Passwortes nichts. Das Terminal-Fenster schließt sich und man kehrt zum Programm-Fenster zurück.

Das was es auch schon – zumindest unter Ubuntu 12.04.

Intern wird die Instruktion Exec() verwendet und auf den folgenden [Befehl](#) – hier im Zusammenhang mit dem Befehl zur Installation des Programms 'bluefish' – zurückgegriffen:

```
hans@linux:~$ /tmp/gambas.1000/10857/xdg-utils/xdg-su -c "apt-get install bluefish"
```

Für die virtuelle Klasse `_Desktop_ScreenSaver` sowie für die Klasse `Desktop` mit den u.a. Methoden erfolgen die Beschreibungen und die Vorstellung relevanter Quelltext-Ausschnitte in eigenen Kapiteln 17.1.1 bis 17.1.5:

- Virtuelle Klasse `_Desktop_ScreenSaver` zur Manipulation des Bildschirmschoners
- `Desktop.Open` (URL As String [,Wait AsBoolean])
- `Desktop.OpenTerminal` ([Dir As String])
- `Desktop.SendKeys` (Keys As String)
- `Desktop.SendMail` (To As String[] [..., Subject As String, Body As String, ...])