

18.7 HSplit und VSplit

Die Komponente *HSplit* ist ein Container, der die eingefügten Komponenten horizontal anordnet; bei 2 Komponenten in einen linken und einen rechten Bereich. Die Komponente *VSplit* dagegen ist ein Container, der die eingefügten Komponenten vertikal anordnet; zum Beispiel in einen oberen und in einen unteren Bereich. Wenn Sie zum Beispiel in die Container-Komponente *VSplit* nicht nur zwei, sondern drei Komponenten einfügen, dann sehen Sie zur Laufzeit 3 Bereiche untereinander, die jeweils durch 2 waagerechte Teiler in der Größe verändert werden können. Die Container *HSplit* und *VSplit* sind zur Laufzeit nicht sichtbar.

Die horizontale oder vertikale *Anordnung* der eingefügten Komponenten wird nur durch die Positionierung der Komponenten auf dem Formular zur *Entwurfszeit* bestimmt! Es gibt keine Methode, die diese Anordnung zur Laufzeit vornimmt. Sie können jedoch zur Laufzeit die Aufteilung der horizontalen Bereiche ändern, indem Sie mit der Maus einen senkrechten Teiler nach links oder rechts verschieben oder die Größe vertikaler Bereiche verändern, wenn Sie einen waagerechten Teiler verschieben. Die Größe der eingefügten Komponenten im Container – zu denen auch der Container Formular gehört – werden nur durch die zur Laufzeit gesetzten Eigenschaften bestimmt, so dass Sie die Komponenten nur wie in der Abbildung 18.7.1.1 positionieren müssen.

Die Verwendung von Containern wie *HSplit* und *VSplit* erfordert ausführlichere Vorüberlegungen zum Formular, als wenn man nur einige Komponenten frei auf einem Formular anordnen möchte. Wenn für Sie feststeht, welche Container Sie einsetzen wollen und welche Komponenten in diese eingefügt werden sollen, dann ist es ein guter Plan, die Anordnung mit Papier und Bleistift – ja, auch ein Füllhalter leistet gute Dienste – festzuhalten. Das wird empfohlen, wenn Sie ein vielfach geschachteltes Formular-Design mit Containern entwickeln, wie es im Beispiel 2 vorgestellt wird.

18.7.1 Beispiel 1 – HSplit

Das folgende Beispiel ist gut geeignet, Sie mit dem Einsatz des Containers *HSplit* vertraut zu machen. Legen Sie ein neues Projekt "HSplit" an. Vergrößern Sie das Formular etwa auf eine Weite von 570 und eine Höhe von 350. Fügen Sie mit *CTRL+E* ein Menü in das Formular ein, indem Sie nach dem Druck auf *Einfügen* den Dialog sofort mit OK verlassen. Nehmen Sie dann den Container *HSplit* als *HSplit1* in das Formular auf und setzen Sie anschließend die beiden Komponenten *DirView1* und *FileView1* in den Container *HSplit1*. Setzen Sie die Eigenschaft *FileView1.ShowDetailed* auf *True*.

Orientieren Sie sich bei der Anordnung der 2 Komponenten im Container *HSplit1* an der folgenden Abbildung, damit die Komponente *DirView1* tatsächlich den linken Bereich einnimmt:

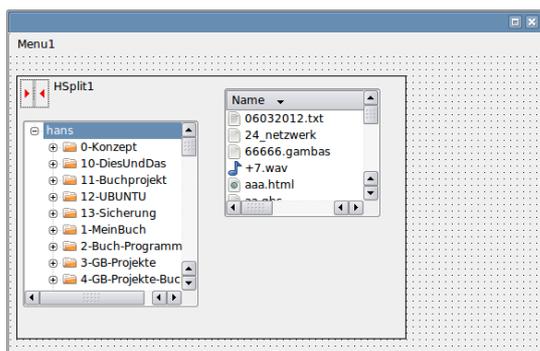


Abbildung 18.7.1.1: Anordnung der Komponenten *DirView1* und *FileView1* im Container *HSplit1* (Entwurf)

Prüfen Sie anschließend, ob die folgenden Eigenschaften der u.a. Komponenten gesetzt sind oder ändern Sie die Werte entsprechend:

- FMain.Arrangement = Fill
- HSplit1.Ignore = False

Nur mit diesen Werten ist garantiert, dass die Komponente *HSplit1* den verbleibenden Platz auf dem Formular vollständig ausfüllt; ohne dem Menü seinen Platz streitig zu machen. Probieren Sie es aus, indem Sie das Programm starten:

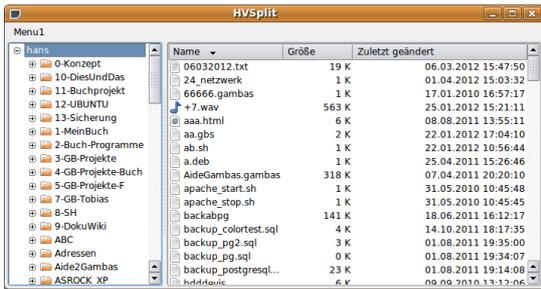


Abbildung 18.7.1.2: Zwei Komponenten im Container HSplit1 (Laufzeit)

Die Verwendung der Komponente VSplit sollten Sie in einem weiteren Projekt erproben.

18.7.2 Beispiel 2 – GBTT

Das folgende Beispiel stellt Ihnen ein Programm zur Verfügung, mit dem Sie auf individuelle Art Ihr Wissen zu Gambas übersichtlich festhalten und pflegen können. Es nutzt neben den vorgestellten Komponenten HSplit und VSplit auch weitere Struktur bestimmende Container und Komponenten. Die folgenden 2 Abbildungen zeigen die Anordnung der einzelnen Komponenten auf dem Formular zur Entwicklungszeit und zur Laufzeit:

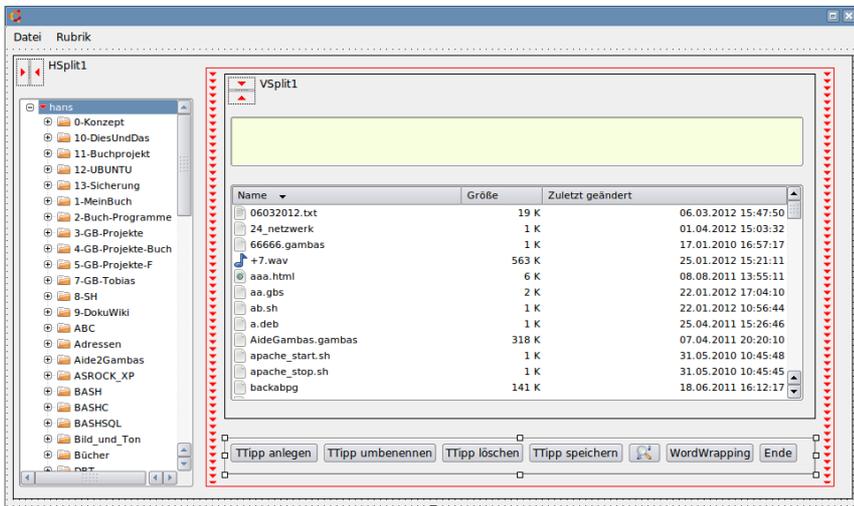


Abbildung 18.7.2.1: Anordnung der Komponenten in mehreren Containern (Entwurfzeit)

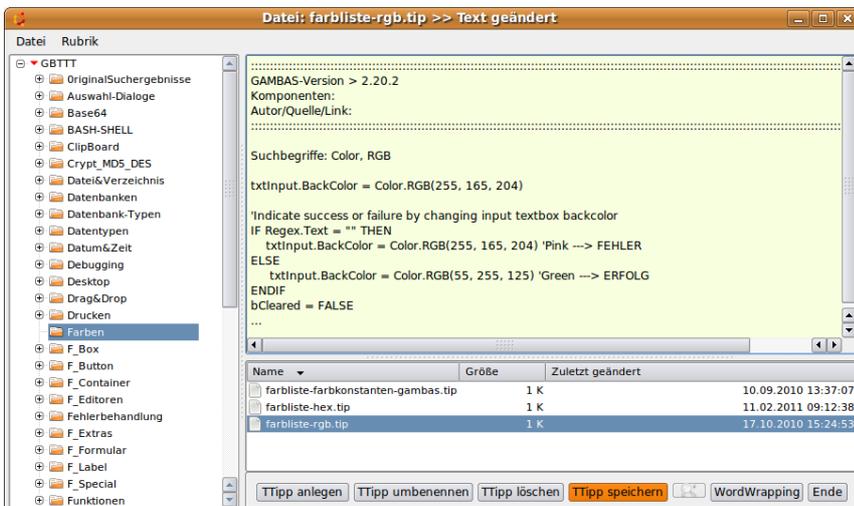


Abbildung 18.7.2.2: Programm GBTT (Gambas-ToolTipp) zur Laufzeit

- Die Komponente HSplit1 vergibt zwei horizontal geteilte Bereiche – links DirView1 und rechts VBox1 – mit einstellbarer Breite durch den senkrechten Teiler.
- Die Komponente VBox1 musste eingefügt werden, da das Panel mit den 7 Button eine feste Höhe und Lage hat und nur die beiden Komponenten TextArea1 (oben mit der grünen Schreibfläche) sowie die Komponente FileView1 im Container VSplit1 durch einen waagerechten Teiler in 2 unterschiedlich große Bereiche geteilt werden.

In den weiteren Abschnitten wird nur auf einige Besonderheiten des Projekts GBTT eingegangen und der entsprechende Quelltext kommentiert.

```
(1) PUBLIC SUB Form_Open()
(2)
(3)   gbttSettings = NEW Settings(User.Home & "GBTTT" & "gbtt.conf")
(4)   HSplit1.Layout = gbttSettings["HSplit/HSplitLayout", "20,80"]
(5)   VSplit1.Layout = gbttSettings["VSplit/VSplitLayout", "70,30"]
(6)   gbttSettings.Read(ME, "Form")
```

Mit den Vorgaben "20,80" = "20%,80%" für den horizontalen Splitter und "70,30" als Teilungsverhältnis für den vertikalen Splitter startet das Programm, da ja beim ersten Start noch keine Konfigurationsdatei existiert. Alle Vorgabe-Werte können als Teilungsverhältnis angegeben werden. Später werden die (absoluten) Werte für das Teilungsverhältnis in Pixeln in die Konfigurationsdatei eingetragen und auch so ausgelesen.

```
(7)   IF FMain.W <= 863 OR FMain.H < 512 THEN
(8)       IF FMain.W < 863 THEN FMain.W = 863
(9)       IF FMain.H < 512 THEN FMain.H = 512
(10)      FMain.Center
(11)   ENDIF ' FMain.W FMain.H?
```

Wenn das Programm mit sehr kleinen Werten für die Breite oder Höhe geschlossen wird, dann werden diese unpraktischen Werte auch in der Ereignisbehandlungsroutine von *Form_Close* gespeichert und ergeben zum Beispiel folgenden Inhalt der Konfigurationsdatei *gbtt.conf*:

```
[FMain/Form]
Geometry=[408,269,377,251]

[HSplit]
HSplitLayout="89,282"

[VSplit]
VSplitLayout="134,45"
```

Beim Auslesen der Formulargrößen nach dem Programmstart – sehen Sie bitte in die Zeile 6 – werden nach einer Werte-Prüfung vernünftige Startwerte zugewiesen.

```
(12) FMain.Center
(13) FMain.Arrangement = Arrange.Fill ' HSplit1 füllt das Formular im Container Formular vollständig aus
(14) VBox1.Ignore = FALSE
(15) HSplit1.Ignore = FALSE
(16) VSplit1.Ignore = FALSE
(17) VSplit1.Expand = TRUE ' VSplit1 füllt den verbleibenden Raum in VBox1, den Panell nicht belegt
(18) Panell.Arrangement = Arrange.Horizontal ' Horizontale Ausrichtung der Button auf dem Panell
(19) Panell.Padding = 10 ' Abstand der 7 Button zum Panell-Rand
(20) Panell.Spacing = 5 ' (Horizontal-)Abstand zwischen den 7 Button
(21) Panell.Height = 41 ' (Original-)Höhe 21 + 2x Panell.Padding = 10
(22) Panell.Border = Border.Sunken ' ... weil es gut aussieht
```

Für eine fehlerfreie Funktion der Anordnung der Container und der Anordnung ausgewählter Komponenten im Container sind alle Zeilen wichtig! Durchdenken Sie in aller Ruhe die Wirkung jeder einzelnen der 10 Zeilen!

```
(23) IF NOT Exist(User.Home & "GBTTT") THEN
(24)   TRY MKDIR User.Home & "GBTTT"
(25)   IF ERROR THEN
(26)       Message.Error("Das Verzeichnis ../GBTTT konnte nicht angelegt werden!")
(27)       RETURN
(28)   ENDIF ' ERROR
(29) ENDIF ' Exist?
(30)
(31) ' ACHTUNG: Das 0 im Wort Original ist eine Null wegen der Sortierung
(32) IF NOT Exist(User.Home & "GBTTT/OriginalSearch") THEN
```

```
(33) TRY MKDIR User.Home &/ "GBTTT/OriginalSearch"  
(34) IF ERROR THEN  
(35)     Message.Error("Das Verzeichnis ../GBTTT/OriginalSearch konnte nicht angelegt werden!")  
(36)     RETURN  
(37) ENDIF ' ERROR  
(38) ENDIF ' NOT Exist?
```

Das Verzeichnis GBTTT sowie die Konfigurationsdatei *gbtt.conf* werden im Home-Verzeichnis beim ersten Programmstart angelegt, wenn sie *nicht* existieren.

```
(39) DirView1.Root = User.Home &/ "GBTTT"  
(40) FileView1.Dir = DirView1.Root  
(41) FileView1.Filter = [ "*.tip", "GAMBAS-Tool-Tipp-Dateien" ]  
(42)  
(43) curZeiger1 = NEW Cursor(Picture["Icon/vonrechts.png"])  
(44) FileView1.Mouse = Mouse.Custom  
(45) FileView1.Cursor = curZeiger1
```

Nur für die Dateiansicht wird ein neuer, benutzer-spezifischer Maus-Zeiger in der IDE als Bild im PNG-Format entworfen, gespeichert und in das Programm eingebunden.

```
(46) btnTTSuchen.Enabled = FALSE  
(47) TextAreal.Wrap = FALSE  
(48) bModifiziert = FALSE  
(49)  
(50) END ' Form_Open
```

Das Word-Wrapping kann ein- und ausgeschaltet werden und ermöglicht eine bessere Lesbarkeit, vor allem bei Tipps, die Sie aus anderen Quellen übernehmen.

```
(1) PUBLIC SUB Form_Close()  
(2)     DIM sToolTippDatei AS String  
(3)  
(4)     gbttSettings["HSplit/HSplitLayout"] = HSplit1.Layout  
(5)     gbttSettings["VSplit/VSplitLayout"] = VSplit1.Layout  
(6)     gbttSettings.Write(ME, "Form")  
(7)     gbttSettings.Save  
(8)  
(9)     FOR EACH sToolTippDatei IN Dir(User.Home & "/GBTTT/OriginalSearch", "**.*")  
(10)        TRY KILL User.Home & "/GBTTT/OriginalSearch/" & sToolTippDatei  
(11)    NEXT ' File  
(12)  
(13) END ' FormClose
```

Hier ist das *.Save* in der Zeile 7 NICHT zwingend erforderlich, weil beim Programmende die Settings-Einstellungen *automatisch* in der Konfigurationsdatei gespeichert werden. Dazu gehören die Werte der aktuellen Teilungsverhältnisse des horizontalen und vertikalen Splitters genauso wie die Formulargröße und dessen Lage-Koordinaten.

Das Programm GBTT verfügt neben einem Menü auch über eine kurze Hilfe im PDF-Format. Über die Suche können Sie gezielt nach Tipps suchen, in denen ein bestimmtes Wort vorkommt. Dazu müssen Sie den Ordner *OriginalSearch* anklicken, damit die Suche über den dann freigegebenen Button mit der kleinen Lupe möglich wird. Alle (temporären) Suchergebnisse finden Sie in der Dateiansicht.