

### 22.11.1 Ausgewählte Klassen der Report-Komponente

In diesem Kapitel werden Ihnen ausgewählte Eigenschaften und Methoden fundamentaler Klassen der Report-Komponente vorgestellt und beschrieben. Die Auswahl erfolgte vor Allem unter dem Aspekt des Einsatzes bei der Erzeugung eines Reports ohne den Report-Designer.

#### 22.11.1.1 Klasse ReportBorder

Die Klasse implementiert ein Objekt, das den Rand eines Report-Steuerelementes darstellt. Die Klassen können Sie erzeugen. Die Klasse verhält sich wie ein schreibgeschütztes statisches Array.

##### 22.11.1.1.1 Eigenschaften der Klasse ReportBorder

Die Klasse *ReportBorder* verfügt über folgende Eigenschaften:

Eigenschaft	Datentyp	Beschreibung
Brush	ReportBrush	Brush stellt das Pinselwerkzeug dar, mit dem man Ränder oder verschiedene Inhalte zeichnen kann.
Left	_ReportBorderSide	Diese Eigenschaft gibt den linken Seitenrand zurück. Einheiten sind: mm, cm, in, ft, px und pt.
Top	_ReportBorderSide	Diese Eigenschaft gibt den oberen Seitenrand zurück.
Right	_ReportBorderSide	Diese Eigenschaft gibt den rechten Seitenrand zurück.
Bottom	_ReportBorderSide	Diese Eigenschaft gibt den unteren Seitenrand zurück.
RoundCorner	_ReportRoundCorner	Diese Eigenschaft setzt das Layout für abgerundete Ecken (BottomLeftCorner, BottomRightCorner, TopLeftCorner und TopRightCorner).
Width	String	Liefert oder setzt die Breite des Rahmens.

Tabelle 22.11.1.1.1 : Eigenschaften der Klasse ReportBorder

Die Klasse besitzt nur eine Methode: ToString() vom Datentyp String. Sie liefert eine formatierte Zeichenkette, welche die komplette Rahmenbeschreibung des Report-Steuerelementes enthält.

So erzeugen Sie ein neues Rahmenobjekt:

```
Dim hReportBorder As ReportBorder
hReportBorder = ReportBorder[sValue As String]
```

Der Wert von *hReportBorder* ist ein String-Array. Sein Inhalt ist eine durch Semikolon getrennte Liste der Eigenschaften des Randes oder eines Teil des Randes. Sie können jeden Rahmeneigenschaften global oder für seine Teile (Top, Right, Left, Bottom und optional für die (abgerundeten) Ecken TopLeftCorner, TopRightCorner, BottomLeftCorner und BottomRightCorner) festlegen.

Der Rand oder ein Teil von ihm wird durch einen Doppelpunkt von seinen einzelnen Eigenschaftswerten getrennt.

- Die Größe in der zulässigen Einheit (mm, cm, in, ft, px, pt) anzugeben. Für den Rahmen werden die Werte der Eigenschaft durch Leerzeichen getrennt.
- Für Ecken können die Werte durch einen Wert oder durch 2 mit Leerzeichen getrennten Werte angegeben werden.
- Der Pinsel wird durch eine ReportBrush-Zeichenkette definiert (optional).

Beispiele:

```
' Der ReportLabel1-Rahmen (links, oben, rechts und unten) wird auf eine Breite von 1 Pixel gesetzt
ReportLabel1.Border = ReportBorder["border:1px"]
```

```
' Der Rand der HBox1 wird auf eine Breite von 2 Pixel und die Farbe schwarz gesetzt.
' Die linke obere und die rechte obere Ecke sind abgerundet.
Report.HBox1.Border = Top:2px #000000;Bottom:2px #000000;Left:2px #000000;Right:2px #000000;
TopLeftCorner:2mm;TopRightCorner:2mm
```

Um einen Rand um ein ReportPanel1 (links, rechts, oben und unten 5 mm, linke obere und untere rechte Ecke zusätzlich abgerundet mit einem Radius von 5 mm) mit linearem Farbverlauf schwarz-rot-gelb (linearer Gradient) zu definieren, benötigen Sie diese Wert-Zuweisung der Eigenschaft ReportPanel1.Border:

```
ReportPanel.Border = ReportBorder["Top:5mm LinearGradient(0,0.04,0,1,[#000000,#FFF00,#FF0000],[0,1,0.5]);Bottom:5mm LinearGradient(0,0.04,0,1,[#000000,#FFF00,#FF0000],[0,1,0.5]);Left:5mm LinearGradient(0,0.04,0,1,[#000000,#FFF00,#FF0000],[0,1,0.5]);Right:5mm LinearGradient(0,0.04,0,1,[#000000,#FFF00,#FF0000],[0,1,0.5]);TopLeftCorner:5mm;TopRightCorner:0mm;BottomRightCorner:5mm;BottomLeftCorner:0mm"]
```

Für einen einfachen schwarzen Rand mit der Stärke 1 Pixel ist der Quelltext wesentlich kürzer:

```
ReportPanel.Border = ReportBorder["Top:1px #000000;Bottom:1px #000000;Left:1px #000000;Right:1px #000000"]
```

### 22.11.1.2 Klasse ReportBoxShadow

Die Klasse repräsentiert einen Schatten eines Report-Steuerelementes vom Typ ReportFrame (→ <http://www.gambaswiki.org/wiki/comp/gb.report2/reportframe>).

Die Klasse können Sie erzeugen:

```
Dim hReportBoxShadow As ReportBoxShadow
hReportBoxShadow = New ReportBoxShadow ( )
```

Die Klasse verhält sich wie ein schreibgeschütztes statisches Array:

```
Dim hReportBoxShadow As ReportBoxShadow
hReportBoxShadow = ReportBoxShadow [ sValue As String ]
```

#### 22.11.1.2.1 Eigenschaften der Klasse ReportBoxShadow

Die Klasse *ReportBoxShadow* verfügt über folgende Eigenschaften:

Eigenschaft	Datentyp	Beschreibung
XOffset	String	Setzt den Offset in x-Richtung oder liest ihn aus. Der Offset bezieht sich auf die rechte untere Ecke des Report-Steuerelementes. Einheiten: mm, cm, in, ft, px und pt.
YOffset	String	Setzt den Offset in y-Richtung oder liest ihn aus. Der Offset bezieht sich auf die rechte untere Ecke des Report-Steuerelementes.
Blur	String	Setzt das Maß für die Unschärfe oder liest es aus. Einheiten: mm, cm, in, ft, px und pt.
Spread	String	Setzt das Maß für die Streuung oder liest es aus. Einheiten: mm, cm, in, ft, px und pt.
Color	Integer	Setzt die Farbe oder liest sie aus.

Tabelle 22.11.1.2.1 : Eigenschaften der Klasse ReportBoxShadow

Beispiel

Die Werte für den Schatten um eine ReportHBox1 werden zuerst einzeln zugewiesen:

```
ReportHBox1.BoxShadow.XOffset = "1mm"
ReportHBox1.BoxShadow.YOffset = "1mm"
ReportHBox1.BoxShadow.Blur = "8px"
ReportHBox1.BoxShadow.Spread = "3px"
ReportHBox1.BoxShadow.Color = "#A5A5A5"
```

Die Werte können auch kompakt zugewiesen werden:

```
ReportHBox1.BoxShadow = ReportBoxShadow["1mm 1mm 10px 3px #A5A5A5"]'[XOffset, Yoffset, Blur, Spread, Color]
```

Sie können den Schatten für die ReportHBox1 aber auch im Dialog im Report-Designer festlegen. Das gelingt schnell und sicher. Zuerst müssen Sie die Eigenschaft *BoxShadow* für ein Report-Steuerele-

ment im Eigenschaftfenster auswählen und über die ButtonBox [...] den Dialog zur *BoxShadow*-Konfiguration aufrufen:



Abbildung 22.11.1.2.1: Dialog zur Festlegung von Schatten-Eigenschaften

Das Ergebnis ist in drei Fällen das Gleiche.

### 22.11.1.3 Klasse ReportPadding

Die Klasse repräsentiert den Innenrand einiger Report-Steuerelemente. Die *Padding*-Eigenschaft gibt eine Instanz an ein *ReportPadding*-Objekt zurück. Dieses Objekt hat 4 Eigenschaften: *Left*, *Right*, *Top* und *Bottom*. Sie definiert den Abstand zwischen dem Inhalt des Report-Steuerelementes und seinem (äußeren) Rand. Es kann jede Seite unabhängig voneinander eingestellt werden. Die Werte können über den entsprechenden Dialog in der IDE oder über Quelltext definiert werden. Die Klasse kann erzeugt werden und verhält sich wie ein statisches Array:

```
Dim hReportPadding As ReportPadding
hReportPadding = ReportPadding [ Value As String ]
```

#### 22.11.1.3.1 Eigenschaften der Klasse ReportPadding

Die Klasse *ReportPadding* verfügt über folgende Eigenschaften:

Eigenschaft	Datentyp	Beschreibung
Bottom	String	Setzt den Wert für den unteren, inneren Rand oder liest ihn aus. Einheiten: mm, cm, in, ft, px und pt.
Left	String	Setzt den Wert für den linken, inneren Rand oder liest ihn aus.
Right	String	Setzt den Wert für den rechten, inneren Rand oder liest ihn aus.
Top	String	Setzt den Wert für den oberen, inneren Rand oder liest ihn aus.

Tabelle 22.11.1.3.1 : Eigenschaften der Klasse ReportPadding

Die Klasse besitzt nur die Methode *ReportPadding.IsActive()*. Die Funktion gibt *True* zurück, wenn die eingestellten *Padding*-Eigenschaften aktiv sind.

#### Beispiel 1

Alle Werte werden auf den gleichen *Padding*-Wert von 2mm gesetzt:

```
ReportLabel1.Padding.Top = "2mm"
ReportLabel1.Padding.Bottom = "2mm"
ReportLabel1.Padding.Left = "2mm"
ReportLabel1.Padding.Right = "2mm"
```

```
ReportLabel1.Padding = ReportPadding["2mm"] ' Alternative 1
ReportLabel.Padding = ReportPadding["Top:2mm;Bottom:2mm;Left:2mm;Right:2mm"] ' Alternative 2
```

#### Beispiel 2

Es wird nur ein oberer, innerer Rand um den *ReportLabel1*-Text gesetzt:

```
ReportLabel1.Padding.Top = "3mm"
ReportLabel1.Padding = ReportPadding["Top:3mm"] ' Alternative
```

Komfortabel gelingen die Festlegungen der Padding-Eigenschaften im Dialog:

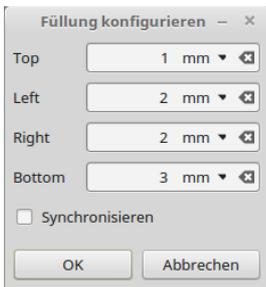


Abbildung 22.11.1.3.1: Dialog zur Festlegung von Padding-Eigenschaften

#### 22.11.1.4 Klasse ReportMargin

Die Padding-Eigenschaft (!) gibt eine Instanz an ein ReportMargin-Objekt zurück. Dieses Objekt hat 4 Eigenschaften: Links, Rechts, Oben und Unten. Sie definiert den Raum zwischen der Grenze eines Report-Steuerelementes und ihrem Report-Container und anderen Report-Steuerelemente, die sich im gleichen Report-Container befinden. Dies ist ein minimaler Platzbedarf. Somit werden zwei in einem Container aufeinander folgende Elemente durch den größten durch die Report-Steuerelemente definierten Abstand oder durch den Abstandswert des Containers getrennt, wenn dieser höher ist. Es kann jede Eigenschaft unabhängig voneinander eingestellt werden. Die Werte können über den entsprechenden Dialog in der IDE oder über Quelltext definiert werden. Die Klasse kann erzeugt werden und verhält sich wie ein statisches Array:

```
Dim hReportPadding As ReportPadding
hReportPadding = ReportPadding [ Value As String ]
```

##### 22.11.1.4.1 Eigenschaften der Klasse ReportMargin

Die Klasse *ReportMargin* hat folgende Eigenschaften:

Eigenschaft	Datentyp	Beschreibung
Bottom	String	Setzt den Wert für den unteren, inneren Abstand oder liest ihn aus. Einheiten: mm, cm, in, ft, px und pt.
Left	String	Setzt den Wert für den linken, inneren Abstand oder liest ihn aus.
Right	String	Setzt den Wert für den rechten, inneren Abstand oder liest ihn aus.
Top	String	Setzt den Wert für den oberen, inneren Abstand oder liest ihn aus.

Tabelle 22.11.1.4.1 : Eigenschaften der Klasse ReportPadding

Die Klasse besitzt nur die Methode *ReportMargin.IsActive()*. Die Funktion gibt True zurück, wenn die eingestellten Padding-Eigenschaften aktiv sind.

##### Beispiel 1

Alle Werte werden auf den gleichen Margin-Wert von 2mm gesetzt:

```
ReportHBox1.Margin.Top = "2mm"
ReportHBox1.Margin.Bottom = "2mm"
ReportHBox1.Margin.Left = "2mm"
ReportHBox1.Margin.Right = "2mm"
```

```
ReportHBox1.Margin = ReportPadding["2mm"] ' Alternative 1
ReportHBox1.Margin = ReportPadding["Top:2mm;Bottom:2mm;Left:2mm;Right:2mm"] ' Alternative 2
```

##### Beispiel 2

Es wird nur ein oberer innerer Abstand um die Report-Steuerelemente im Report-Container ReportHBox1 gesetzt:

```
ReportHBox1.Margin.Top = "3mm"
ReportHBox1.Margin = ReportPadding["Top:3mm"] ' Alternative
```

Komfortabel gelingen die Festlegungen der Eigenschaften für ein Margin-Objekt im Dialog:

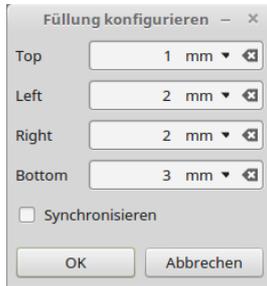


Abbildung 22.11.1.4.1: Dialog zur Festlegung von Padding-Eigenschaften für ein Margin-Objekt

### 22.11.1.5 Klasse ReportBrush

Die Klasse repräsentiert das Pinselwerkzeug, mit dem Sie Ränder oder verschiedene Inhalte zeichnen können. Das kann eine einzelne Farbe sein, aber auch ein Farbverlauf oder ein Bild. Die Klasse können Sie erzeugen und sie verhält sich wie ein schreibgeschütztes statisches Array. Der folgende Quelltext erzeugt ein neues ReportBrush-Objekt aus einer Pinselbeschreibung in *sValue* und gibt es zurück:

```
Dim hReportBrush As ReportBrush
hReportBrush = ReportBrush [ sValue As String ]
```

#### 22.11.1.5.1 Eigenschaften der Klasse ReportBrush

Die Klasse *ReportBrush* besitzt folgende statische Eigenschaften:

Eigenschaft	Datentyp	Beschreibung
Function Color ( iColor As Integer )	ReportBrush	Setzt die Pinselfarbe.
Function Image ( hImage As Image )	ReportBrush	Setzt ein Bild als Pinselwerkzeug.
LinearGradient ( X As Float, Y As Float, X2 As Float, Y2 As Float, aColor As Integer[], aPos As Float[] )	ReportBrush	Setzt einen linearen Farbverlauf als Pinselwerkzeug.
RadialGradient ( X As Float, Y As Float, Radius As Float, X2 As Float, Y2 As Float, aColor As Integer[], aPos As Float[] )	ReportBrush	Setzt einen radialen Farbverlauf als Pinselwerkzeug.

Tabelle 22.11.1.5.1 : Eigenschaften der Klasse ReportBrush

#### Beispiel 1

```
' Setzt eine Zeichenfarbe
MyBrush = ReportBrush["#00FF00"]
```

```
' Setzt ein Bild als Zeichenwerkzeug
MyBrush = ReportBrush["image(\"icon.png\")"] oder MyBrush = ReportBrush["image(icon.png)"]
```

```
' Setzt einen linearen Farbverlauf
' Alle Werte liegen zwischen 0 und 1 - auch für die Größe und den Gradientenstop.
' Syntax:
' relleft,reltop,relwidth,relheight,[colorstop1,colorstop2,...],[colorstopRelPos1,colorstopRelPos2,...])
MyBrush = ReportBrush["LinearGradient(0,0,1,1,[#000000,#FFFFFF,#0000FF],[0,1,0.5)"]
```

```
' Setzt einen radialen Farbverlauf
' Alle Werte liegen zwischen 0 und 1 - auch für die Größe und den Gradientenstop.
' Syntax:
' relCenterX,relCenterY,relRadius,relFX,relFY,[colorstop1,colorstop2,...],[colorstopRelPos1,colorstopRelPos2,...])
MyBrush = ReportBrush["RadialGradient(0.53,0.53,0.5,0.88,0.18,[#000000,#FFFFFF,#0000FF],[0,1,0.5)"]
```

Ein Beispiel für die Verwendung eines Pinsels mit linearem Farbverlauf sehen Sie in der Abbildung 22.11.1.5.2.

Wesentlich komfortabler gelingt Ihnen ein Farbverlauf im entsprechenden Dialog. Zuerst müssen Sie die Eigenschaft *Brush* für ein Report-Steuerelement im Eigenschaftfenster auswählen und über die ButtonBox [...] den Dialog zur Brush-Konfiguration aufrufen:

- (1) In der Combobox 'Linearer Gradient' auswählen
- (2) Gradient-Haltepunkt hinzufügen (Klick auf das +)
- (3) Doppelklick auf den ersten Gradient-Haltepunkt → Farbe Schwarz mit OK auswählen
- (4) Doppelklick auf den zweiten Gradient-Haltepunkt → Farbe Rot mit OK auswählen
- (5) Doppelklick auf den dritten Gradient-Haltepunkt → Farbe Gelb mit OK auswählen
- (6) Den Anfasser \* im Farbfeld (links oben) anklicken und in die rechte untere Ecke ziehen
- (7) Den nun sichtbaren zweiten Anfasser in die linke untere Ecke ziehen
- (8) Den Anfasser \* aus der rechten unteren Ecke auf die linke obere Ecke ziehen

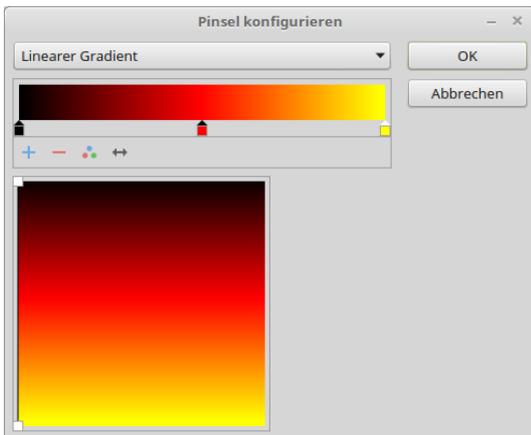


Abbildung 22.11.1.5.1: ReportPanel1-Rand

Mit OK übernehmen Sie den konfigurierten Pinsel mit dem vertikalen Farbverlauf Schwarz-Rot-Gelb:



Abbildung 22.11.1.5.2: ReportPanel1-Rand