

Das Package Multis

Installation

TMultiPanel

TMultiButton

TMultiButtonStylemanager

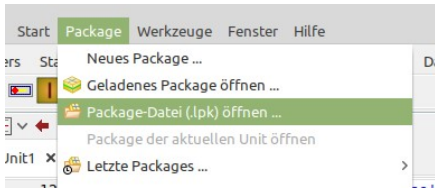
TMultiplexSlider

TMultiSeperator

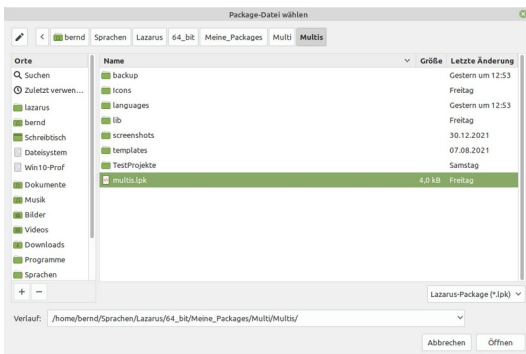
Installation

Das Package befindet sich in folgendem Githubkonto: <https://github.com/wennerer/Multis>

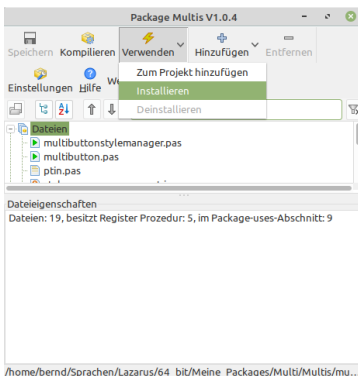
Nachdem das Package geklont oder heruntergeladen wurde kann es in Lazarus installiert werden. Dazu Lazarus öffnen, dort unter Package den Punkt Package-Datei (.lpk) öffnen... anklicken.



Jetzt in den Ordner Multis navigieren und dort die Datei multis.lpk auswählen.

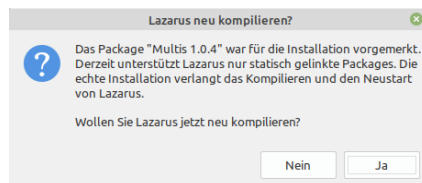


Es öffnet sich folgendes Fenster:

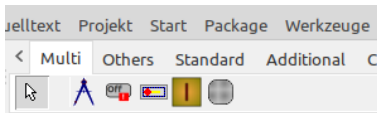


Hier auf Verwenden und dann Installieren.

Diesen Dialog mit Ja bestätigen:



Nun befindet sich ein neuer Reiter Multi in der Palettenauswahl.



TMultiPanel

Eigenschaften

Action	: Die dem Steuerelement zugeordnete Aktion
Align	: Gibt die Platzierung des Steuerelements innerhalb seines übergeordneten Elements an
Anchors	: Der Satz von Ankerdefinitionen für dieses Steuerelement
Autosize	: Ermöglicht die automatische Anpassung der Größe des Steuerelements an seinen Inhalt
BidiMode	: Anpassung (von Textsteuerelementen) in bidirektionalen Leseumgebungen
BorderSettings	: Die Eigenschaften des Randes
BorderSettings.Between	: Der Raum zwischen Innen- und Außenrand
BorderSettings.InnerColor	: Die Farbe des inneren Randes
BorderSettings.InnerWidth	: Die Dicke des inneren Randes
BorderSettings.OuterColor	: Die Farbe des äußeren Randes
BorderSettings.OuterWidth	: Die Dicke des äußeren Randes
BorderSpacing	: Bestimmt den inneren und äußeren Rahmenabstand für dieses Steuerelement
Caption	: Der Text den der Benutzer in das Panel schreibt
CaptionAlignment	: Ausrichtung des Textes in der Caption (Links,Mitte,Rechts)
CaptionHorMargin	: Der horizontale Abstand des Textes im Textrechteck (nur wirksam mit taLeftJustify)
CaptionLayout	: Ausrichtung des Textes in der Caption (Oben,Mitte,Unten)
CaptionVerMargin	: Der vertikale Abstand des Textes im Textrechteck (nur wirksam mit tlTop)
CaptionWordbreak	: Ermöglicht einen Zeilenumbruch in der Caption
ColorEnd	: Die Endfarbe des Panels (für Farbverlauf)
ColorGradient	: Die Richtung des Farbverlaufs
ColorStart	: Die Startfarbe des Panels (für Farbverlauf)
Constraints	: Die minimale und maximale Breite und Höhe für das Steuerelement
Cursor	: Die Form des Mauszeigers, wenn sich die Maus über diesem Steuerelement befindet
DoubleBuffered	: Ermöglicht das Reduzieren des Flimmerns in der Lackierung des Steuerelements
DragCursor	: Die Cursorform, die angezeigt wird, während das Steuerelement gezogen wird
DragKind	: Der Vorgang, wenn das Steuerelement gezogen wird - Drag or Dock
DragMode	: Ermöglicht dem Benutzer, das Steuerelement zu ziehen
DrawACustomPanel	: Öffnet einen Editor in dem man ein Panel zeichnen kann
DropDownMenu	: Die Eigenschaften der DropDownfunktion
DropDownMenu.Active	: Aktiviert die DropDown-Funktion
DropDownMenu.Compressed	: Eigenschaften des komprimierten Panels
DropDownMenu.Compressed.Active	: Macht die Auswahl zum Startwert
DropDownMenu.Compressed.Height	: Die Höhe des komprimierten Panels
DropDownMenu.Compressed.Width	: Die Breite des komprimierten Panels
DropDownMenu.Direction	: Die Ausklapprichtung
DropDownMenu.Hotspot	: Legt den Bereich fest in dem ein Klick wirkt, nur aktive mit DropDownMenu.Active und trPinned (nur zur Laufzeit!)

<code>DropDownMenu.Speed</code>	: Die Zeichengeschwindigkeit (Timer Intervall)
<code>DropDownMenu.Step</code>	: Der Zeichenschritt (in Pixel)
<code>DropDownMenu.Stretched</code>	: Eigenschaften des gedehnten Panels
<code>DropDownMenu.Stretched.Active</code>	: Macht die Auswahl zum Startwert
<code>DropDownMenu.Stretched.Height</code>	: Die Höhe des gedehnten Panels
<code>DropDownMenu.Stretched.Width</code>	: Die Breite des gedehnten Panels
<code>DropDownMenu.Trigger</code>	: Der Auslöser
Font	: Die Schrift die für die Textanzeige in diesem Panel verwendet werden soll
Height	: Die vertikale Größe des Steuerelements
HelpContext	: Die ID für die kontextbezogene Hilfe zu diesem Steuerelement
HelpKeyword	: Das Schlüsselwort für die kontextbezogene Hilfe zu diesem Steuerelement
HelpType	: Legt fest, ob die kontextsensitive Hilfe nach numerischer ID oder Schlüsselwort ausgewählt wird
Hint	: Der Text, der im Hinweisenster für das Steuerelement angezeigt werden soll
ImageIndex	: Der Index eines Bildes in einer ImageList
ImageLeft	: Die Koordinate der linken Ecke des Bildes
Images	: Eine Liste zum Einfügen von Bildern
ImageTop	: Die Koordinate der oberen Ecke des Bildes
ImageWidth	: Die einmalige Breite aller Bilder in der Liste
<code>Left</code>	: Die Clientkoordinate des linken Rands des Steuerelements
<code>RndRctRadius</code>	: Eckendurchmesser wenn geometrische Form ist RoundRect
<code>Style</code>	: Die geometrische Form des Panels
<code>Top</code>	: Die Clientkoordinate des oberen Rands des Steuerelements
Visible	: Ermöglicht das Ein- oder Ausblenden des Steuerelements und aller seiner untergeordneten Elemente
Width	: Die horizontale Ausdehnung des Steuerelements
<code>Appear</code>	: läßt das Panel erscheinen (nur zur Laufzeit!)
<code>Disappear</code>	: läßt das Panel verschwinden (nur zur Laufzeit!)
<code>AnimationSpeed</code>	: Geschwindigkeit für Appear bzw. Disappear (default 0,05) (nur zur Laufzeit!)
<code>ParentAsBkgrd</code>	: Hintergrund des Panels nimmt die Farbe des Parent an (nur zur Laufzeit!)

Ereignisse

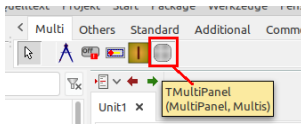
OnChangeBounds	: Handler für eine Änderung der Grenzen des Steuerelements
OnClick	: Benachrichtigungshandler für Mausklicks
OnCompressed	: Handler wenn das Panel komprimiert ist, nur aktive wenn DropDownMenu.Active
OnDragDrop	: Dieser Handler bestimmt die Aktion beim Ablegen auf diesem Steuerelement in einem Drag-Drop-Vorgang
OnDragOver	: Ereignishandler für ein Steuerelement, das über dieses Steuerelement gezogen wird
OnEndDrag	: Benachrichtigungshandler für das Ende eines Ziehvorgangs
OnEnter	: Handler für die Steuerung, die den Fokus erhält
OnExit	: Handler für die Steuerung, die den Fokus verliert;
OnKeyDown	: Handler für gedrückte Tastaturtaste
OnKeyPress	: Handler für ein vom Benutzer eingegebenes Zeichen
OnKeyUp	: Handler für Tastaturtaste freigegeben
OnMouseDown	: Ereignishandler für das Drücken der Maustaste
OnMouseEnter	: Ereignishandler für das Betreten des Bereichs des Steuerelements mit der Maus
OnMouseLeave	: Ereignishandler für Maus, die den Bereich des Steuerelements verlässt
OnMouseMove	: Ereignishandler für die Mausbewegung innerhalb des Steuerelements
OnMouseUp	: Ereignishandler für das Loslassen der Maustaste
OnStartDrag	: Ereignishandler für den Start eines Ziehvorgangs
OnStretched	: Handler wenn das Panel ausgeklappt ist, nur aktive wenn DropDownMenu.Active

Öffentliche Prozeduren

```
procedure MouseMove({%H-}Shift: TShiftState; X, Y: Integer);override;
procedure MouseDown({%H-}Button: TMouseButton;{%H-}Shift: TShiftState; X, Y: Integer);override;
procedure MouseUp({%H-}Button: TMouseButton; {%H-}Shift: TShiftState; {%H-}X, {%H-}Y: Integer);override;
procedure LoadFromFile(aFileName: string);
procedure InvalidateBackground;
procedure ParentInputHandler({%H-}Sender: TObject; Msg: Cardinal);
procedure Notification(AComponent: TComponent;Operation: TOperation); override;
constructor Create(AOwner: TComponent); override;
destructor Destroy; override;
procedure MouseEnter; override;
procedure MouseLeave; override;
procedure Paint; override;
```

Beschreibung

Das MultiPanel findest du im Reiter Multis

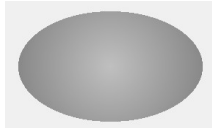


Die Form des MultiPanels lässt sich mit der Eigenschaft [Style](#) beeinflussen.

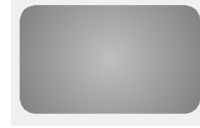
mpsRect:



mpsEllipse:



mpsRoundRect:

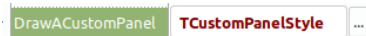


Der Eckenradius lässt sich mit [RndRctRadius](#) einstellen.
Standardeinstellung ist 40

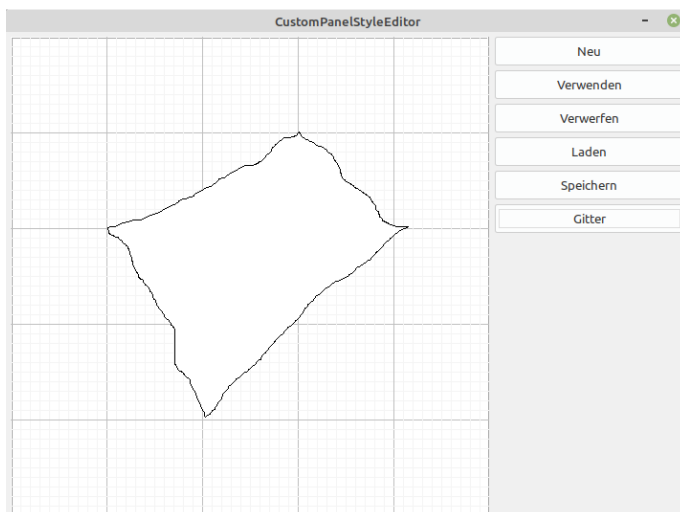
mpsCustom:



Bei mpsCustom ist standardmäßig ein Dreieck hinterlegt. Um ein benutzerdefiniertes Panel zu zeichnen muss man auf auf die 3 Punkte hinter [DrawACustomPanel](#) klicken.



Es öffnet sich ein Eigenschaftseditor:



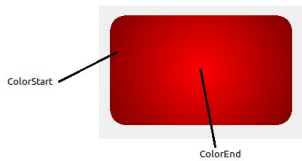
Klickt man jetzt auf Neu kann man einfach mit der Maus die Form des MultiPanels zeichnen. Klickt man auf Verwenden wird die MultiPanel-Form übernommen. Mit Verwerfen wird die MultiPanel-Form nicht übernommen und der Editor geschlossen. Mit Speichern kann man eine gezeichnete Form abspeichern und mit Laden wieder holen. Gitter blendet ein Hilfgitter ein das einem eventuell beim Zeichnen helfen kann.

Bedenke es muss mpsCustom eingestellt sein!

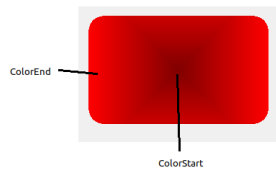
Zur Laufzeit lassen sich auch mit [LoadFromFile](#) vorab gespeicherte MultiPanel's laden.

Um die Farbe des MultiPanels zuverändern benötigst du die Eigenschaften [ColorStart](#), [ColorEnd](#) und [ColorGradient](#). Um ein einfarbiges MultiPanel zu bekommen muss ColorStart und ColorEnd gleich sein. Ansonsten bestimmt die Zusammensetzung aus den drei Eigenschaften das Aussehen.

gcSpread:



gcRadiant:

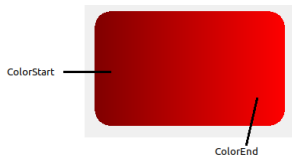


gcAlternate:

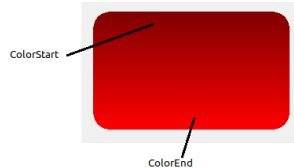


Setzt abwechselnd ein Pixel je auf Start- und EndColor.

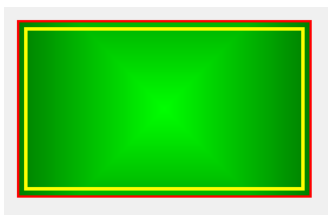
gcHorizontal:



gcVertical:



Wenn man den Rand hervorheben möchte stehen die Eigenschaften des [BorderSettings](#) zur Verfügung.



BorderSettings	
Between	7
InnerColor	clYellow
InnerWidth	3
OuterColor	clRed
OuterWidth	3

Um einen Rand zu erzeugen muss man einfach eine Farbe auswählen. Will man keinen Rand verwendet man clNone.

[BorderSettings.Between](#)

: Der Raum zwischen Innen- und Außenrand

[BorderSettings.InnerColor](#)

: Die Farbe des inneren Randes

[BorderSettings.InnerWidth](#)

: Die Dicke des inneren Randes

[BorderSettings.OuterColor](#)

: Die Farbe des äußeren Randes

[BorderSettings.OuterWidth](#)

: Die Dicke des äußeren Randes

Die Eigenschaften [Appear](#), [Disappear](#) und [AnimationSpeed](#) können nur zur Laufzeit gesetzt werden!

Um ein unsichtbares MultiPanel erscheinen zu lassen benutzt man die Eigenschaft [Appear](#).

Beispiel-Code:

```
procedure TForm1.MultiButton1Click(Sender: TObject);
begin
    MultiPanel1.Appear:= true;
end;
```

Um ein sichtbares MultiPanel verschwinden zu lassen benutzt man die Eigenschaft [Disappear](#).

Beispiel-Code:

```
procedure TForm1.MultiButton2Click(Sender: TObject);
begin
    MultiPanel1.Disappear:= true;
end;
```

Mit der Eigenschaft [AnimationSpeed](#) lässt sich die Geschwindigkeit des Erscheinen bzw. Verschwindens beeinflussen.

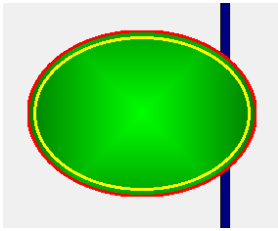
Standartwert ist 0,05. Je kleiner der Wert desto langsamer geht die Animation vonstatten. Bei einem Wert von 0,001 geht es schon sehr langsam.

Beispiel-Code:

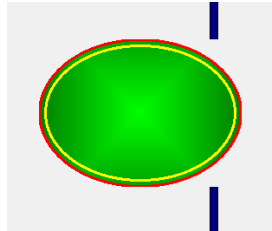
```
MultiPanel1.AnimationSpeed:= 0.001;
```


Wählt man als geometrische Form (Eigenschaft Style) etwas anderes als `mpsRect` so wird ein Teil des Hintergrundes des `MultiPanels` sichtbar. Diese nun sichtbaren Ecken nehmen die im Parent gesetzte Farbe an. Befinden sich im Parent zum Beispiel selbst gezeichnete Linien werden diese auch gezeigt. Dies geschieht da standardmäßig die Eigenschaft `ParentAsBkgrd` auf `true` gesetzt ist.

`ParentAsBkgrd := true`



`ParentAsBkgrd := false`

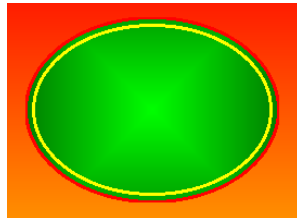


Diese Einstellung macht vor allem Sinn wenn der Parent seine Größe ändert. Denn dann wird zum Beispiel hier die gezeichnete Linie nicht richtig skaliert.

Besitzt der Parent einen Farbverlauf besteht die Möglichkeit die Skalierungsproblematik durch Aufruf der Prozedur `InvalidateBackground` auszugleichen.

Beispiel-Code:

```
procedure TForm1.FormChangeBounds(Sender: TObject);
begin
  MultiPanel1.InvalidateBackground;
end;
```



Um ein **DropDown** Menü (Hamburgermenü) zu erzeugen stellt man zuerst die Eigenschaft **DropDownMenu.Active** auf **true**.

DrawACustomPanel	TCustomPanelStyle
DropDownMenu	(TDDMenu)
Active	<input checked="" type="checkbox"/> [True]
Compressed	(TComp)
Direction	LeftTop_RightBottom
Speed	3
Step	2
Stretched	(TStre)
Trigger	trHover

Das MultiPanel zeigt nun den Komprimierten Zustand. Die Positionierung kann mit der Maus oder den Eigenschaften **Left** bzw. **Top** gemacht werden (natürlich können auch Anker gesetzt werden). Die Größe kann man einfach mit der Maus ziehen oder mit den Eigenschaften **DropDownMenu.Compressed.Height** bzw. **DropDownMenu.Compressed.Width** zuweisen. Passt die Größe schaltet man mit den Eigenschaften **DropDownMenu.Stretched.Active** bzw. **DropDownMenu.Compressed.Active** auf den ausgedehnten Zustand. Nun kann ebenfalls die gewünschte Größe durch ziehen mit der Maus oder den Eigenschaften **DropDownMenu.Stretched.Height** bzw. **DropDownMenu.Stretched.Width** eingestellt werden. Es empfiehlt sich in diesen

Zustand die gewünschten Kinder-Controls (Button's etc.) zu platzieren. Die Richtung nach der das MultiPanel ausklappt wird durch die Eigenschaft **DropDownMenu.Direction** bestimmt. Zur Auswahl stehen folgende Möglichkeiten:

TDirection = (LeftTop_RightBottom, RightTop_LeftBottom, LeftBottom_RightTop, RightBottom_LeftTop)

Die Geschwindigkeit des Ausklappens kann man durch die Eigenschaften **DropDownMenu.Speed** und **DropDownMenu.Step** beeinflussen. Hinter Speed versteckt sich das Timer-Intervall mit dem das Ausklappen aufgerufen wird. Zum Verlangsamen erhöht man dieses Wert bis zur gewünschten Geschwindigkeit. Mit Step lässt sich die Anzahl der Pixel einstellen die je Intervall zusätzlich gezeichnet werden. Will man also schneller Ausklappen erhöht man also den Wert bei Step.

Mit **DropDownMenu.Trigger** bestimmt man den Auslöser für das Ausklappen. Es stehen folgende Möglichkeiten bereit:

TTrigger = (trClick, trHover, trPinned)

Bei trClick muss in das Panel geklickt werden, bei trHover reicht es mit der Maus darüber zu fahren. Bei trPinned muss in das MultiPanel geklickt werden, das Einklappen erfolgt jedoch nur wenn in einem festlegbaren Hotspot (**DropDownMenu.Hotspot**) geklickt wird.