

[Properties](#) [Methods](#) [Events](#) See Also: [TMPPane](#)

TTRIPANE

Ancestor: TCustomPanel

TTripane is versatile panel-type component that provides 3 sub-panes with a variety of properties and functions useful for segmenting panels into distinct areas. This permits regions of an application to be more easily accessed for specific properties, either at design or runtime. Using a Tripane is often much more efficient and organized than using separate panels. Experimentation with this control is the best way to understand it.

A Tripane consists of

Client Pane -- for containing any primary controls or data within the Tripane. As with the other sub-panes below, many properties of TCustomPanel are published.

Header Pane -- useful for toolbars, title areas, function dialogs, rulers or other application features where variable heights, borders and other properties might be useful.

Menu Pane -- is a sub-pane which may "pop-open" over the client area from either the left, top, right or bottom of the client area. This pane is useful for secondary data or navigation controls, and generally useful as a pop-open menu area and specifically useful as a container for SlickTool's TMenuBar control.

A mouse-triggered "stripe zone" is used to [PopOpen](#) or [PopClose](#) the Menu area. This can be color coded, or simply widened to provide a noticeable "trigger area." This feature can be enabled or disabled with the [MouseTrigger](#) property.

To enable or disable Tripane's advanced mouse tracking, see: [MouseEventsEnabled](#)

Included Classes:

TTriPanePopEvent = procedure(Sender:TObject;Open:boolean) of object
TMenuAlignment = (mnTop, mnBottom, mnLeft, mnRight);
TBevels = exNone, exThickRaised, exThickSunk, exThinSunk, exSimpleBorder);
TExclusionCoordinates = set of byte;
TMPPane
TTriPane

[Properties](#) [Events](#) [Overview](#)

TTripane Methods

[Create](#)

[Destroy](#)

[Exclude](#)

[IncludeAll](#)

[PopOpen](#)

[PopClose](#)

See also: methods of ancestor class: [TCustomPanel](#)

Constructor Create(AOwner:Tcomponent)

Creates a new TTriPane.

Destructor Destroy

Destroys a TriPane. Use **Free** instead.

Procedure Exclude(offset:integer;pixels:TExclusionCoordinates);virtual;

Use Exclude to exclude a pixel range from tracking for mouse open and close. This is useful if the mousable "Trigger Area" for a Tripane menu sub-pane is too close to another control and might be inadvertently triggered.

Exclusions work only in one axis. The axis in which they function is defined by the AlignMenu property. If AlignMenu is mnLeft or mnRight, Exclude will exclude ranges of y coordinates. If AlignMenu is mnTop or mnBottom, Exclude will exclude ranges of x coordinates. This is to facilitate the PopOpen action not being triggered when the mouse is in specific positions adjacent to the Stripe area, where it otherwise would be triggered, while maintaining normal trigger sensitivity for all coordinate ranges not excluded.

Offset is the amount of pixels to add to each value in the the set. Offset is necessary in order to reach coordinate values above 255.

Pixels are TExclusionCoordinates. As a Delphi set may only contain 256 elements, it is necessary to specify an offset to enable the use of set values (coordinate values) above 255.

Thus, in order to exclude the pixel ranges of
10-20, 267-295, and 400,403, and 405, one would call:

```
TriPane1.Exclude(0,[10..20]);  
TriPane1.Exclude(267,[0..28]);  
TriPane1.Exclude(400,[0,3,5]);
```

respectively.

The Exclusions are cumulative, in that they are stored by the TriPane in a TList, so multiple calls of Exclude will not overwrite previous calls.

Note: Use IncludeAll to eliminate all defined exclusion sets.

Procedure IncludeAll

Eliminates all Exclusion sets. Use if the TriPane is resized or if it becomes necessary to move exclusion zones.

Procedure PopClose

Call PopClose to close the Menu area of a Tripane

Procedure PopOpen

Call PopOpen to open the Menu area of a Tripane.

The size of the opened Menu is determine by PopOpenSize property.

[Methods](#) [Events](#) [Overview](#)

TTripane Properties

[AlignMenu](#)
[Client](#)
[ContinuousAlign](#)
[GrabBorder](#)
[Header](#)
[HeaderPaneHeight](#)
[Menu](#)
[MouseEventsEnabled](#)
[MouseTrigger](#)
[Open](#)
[PaneBevelInner](#)
[PaneBevelOuter](#)
[PaneBevelWidth](#)
[PaneBorderStyle](#)
[PaneColor](#)
[PopOpenSize](#)
[Proximity](#)
[SlideIncrement](#)
[Stripe](#)
[StripeColor](#)
[StripeWidth](#)

Derived from TCustomPanel

Align
BevelInner
BevelOuter
BevelWidth
BorderStyle
BorderWidth
Visible

See also: properties of ancestor class: TCustomPanel

property AlignMenu: TMenuAlignment

Read or set the Menu pane's alignment in the TriPane.
The menu area can be positioned on any of the four sides of the TriPane.

property Client: TMPPane

A reference to the Client area pane.

property ContinuousAlign

If this property is true, the Client area will continuously realign itself as (if) the Menu pane slides open and closed.

In order to prevent control misalignment, the Client area will realign itself even if the property is set to false, if the Tripane is Open and the Tripane is resized.

property GrabBorder: boolean

Use to set a preset borderwidth of 2 to the TriPane in order to assist in grabbing it at design-time. This value is ignored at run-time.

property Header: TMPPane

A reference to the Header area pane.

property BevelingEx: TBevels

Read or set extended beveling options.

These bevels apply to the Tripane as a whole. BevelingEx implements WS_EX_ window styles.

property HeaderPaneHeight

Controls the height of the Header pane area.

property Menu: TMPPane

A reference to the Menu area pane.

property MouseEventsEnabled :boolean

This property should be set to false to recursively disable all MouseMove tracking for subcomponents of the TriPane.

In order to track the mouse's movements, TTriPane recursively sets all MouseMove events of its child controls in order to determine the mouse position relative to the parent TriPane. If it becomes necessary at runtime to change the parent of an eventual child of a TriPane, such as when undocking a control or changing a control's parent, one must set MouseEventsEnabled: = false **before** changing the parent or undocking the control, or one may receive an access violation as soon as the mouse pointer goes over the control that has had its parent changed.

Immediately after changing the parent, one should set MouseEventsEnabled:=true to reenble the tracking ability of the TriPane.

property MouseTrigger

Use to set whether the mouse will trigger PopOpens and PopCloses.

Note: this **is not** the same as MouseEventsEnabled.

property Open:boolean

Reads whether the TriPane menu area is open or not.

Do not confuse this with the Open parameter of TTriPanePopEvent.

property PaneBevelInner:TPanelBevel

Design-only. Use to set the value of the highlighted pane at *design-time only*. To set at runtime, specify bevel style for the client, header or menu properties of TTripane.

eg. Tripane1.Client.bevelinner := bvRaised;

property PaneBevelOuter:TPanelBevel

Design-only. Use to set the value of the highlighted pane at *design-time only*. To set at runtime, specify bevel style for the client, header or menu properties of TTripane.

eg. Tripane1.Client.bevelouter := bvRaised;

property PaneBevelWidth:integer

Design-only. Use to set the value of the highlighted pane at *design-time only*. To set at runtime, specify bevel width for the client, header or menu properties of TTripane.

eg. Tripane1.Client.bevelwidth := 2;

property PaneBorderStyle:TBorderStyle

Design-only. Use to set the value of the highlighted pane at *design-time only*. To set at runtime, specify borderstyle for the client, header or menu properties of TTripane.

eg. Tripane1.Client.borderstyle := bsSingle

property PaneColor:TColor

Design-only. Use to set the Color of the highlighted pane at *design-time only*. To set at runtime, specify Color for the client, header or menu properties of TTripane.

eg. Tripane1.Client.Color := clgray

property PopOpenSize

Sets the size of the Menu pane area when popped-open.

property Proximity

Controls the pixel range within the appropriate edge of the Client area after which the Menu area will pop open.

property SlideIncrement

Change this value to control the increment at which the Menu "slides" open. A value less than zero specifies that the Menu should pop-open instantly.

property Stripe:TPanel

A reference to the Stripe area of the TriPane. The stripe resides between the Menu pane and the Client pane.

property StripeColor

Use to set/read the color of the Stripe area.

property StripeWidth:byte

Use to set/read the width (or height), in pixels, of the Stripe region.

[Properties](#) [Methods](#) [Overview](#)

TTripPane **Events**

[OnPopped](#)

[OnPopping](#)

Derived from TCustomPanel

[OnResize](#)

See also: events of ancestor class: [TCustomPanel](#)

property OnPopping:TTriPanePopEvent

Event called before the TriPane pops open or pops closed.

The open parameter specifies whether the Tripane's Menu is opening (true) or closing (false).

Open = true	Tripane is Opening
Open = false	Tripane is Closing.

property OnPopped::TTriPanePopEvent

Event that is called after the TriPane has popped open or popped closed.

The open parameter specifies whether the Tripane's Menu is closed (false) or open (true).

Open = true	Tripane is Opened.
Open = false	Tripane is Closed.

Properties

TMPPane

TMPPane is the class used for one of the three subpanes of a Tripane. Again, the visible member panes of TTripane can be accessed:

Tripane1.Header

Tripane1.Menu

Tripane1.Client

TMPPane Properties

Derived from TCustomPanel

align
bevelInner
bevelOuter
bevelWidth
borderWidth
borderStyle
color

TripanePopEvent

type **TTripanePopEvent** = **procedure** (Sender:TObject;Open:boolean) **of object**;

The Open parameter specifies whether the Tripane is opening or closing.

Open = true for opening/open

Open = false for closing/closed.

See also [OnPopped](#) and [OnPopping](#) of TTriPane

Type TMenuAlignment = (mnTop,mnBottom,mnLeft,mnRight)

TMenuAlignment specifies where the Menu appears in a Tripane.

```
type TBevels = exNone, exThickRaised, exThickSunk, exThinSunk, exSimpleBorder
```

TBevels are special additional bevel types of: BevelingEx which may be applied to a Tripane (in addition to standard bevel values for TCustomPanel Bevelinner or Bevelouter).

Values:

- exNone
- exThick
- exRaised
- exThickSunk
- exThinSunk
- exSimpleBorder

Note: TBevels values may only be applied to the entire TTripane. Use specific Bevel properties of TCustomPanel for individual sub-panes.

e.g., Tripane1.Client.Bevelinner := bvRaised; //sets the BevelInner of the Client pane

TExclusionCoordinates = **set** of byte;

Company Information

SlickTools is the components products division of InfoMatters, LLC.

SlickTools
1111 Arlington Blvd.
Suite 519
Arlington, VA 22209

www.slicktools.com
SlickTools@aol.com

SLICKTOOLS ORDERFORM

