

**tcpalette\_gc**

COLLABORATORS

	TITLE : tcpalette_gc		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY		January 23, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>tcpalette_gc</b>	<b>1</b>
1.1	tcpalette_gc.doc . . . . .	1
1.2	/" . . . . .	1
1.3	tcpalette.gc/--datasheet-- . . . . .	1

## Chapter 1

# tcpalette\_gc

### 1.1 tcpalette\_gc.doc

```
--datasheet-- ()
```

### 1.2 /"

To Do

- \* Add TCPALETTE\_PenNameArray (UBYTE \*\*) Make paletter gadget look like gadget in Palette Prefs
- May need scroller ?

### 1.3 tcpalette.gc/--datasheet--

NAME

tcpalette.gadget -- True color palette BOOPSI gadget

SUPERCLASS

gadgetclass

REQUIRES

mlr\_ordered.pattern  
bevel.image

DESCRIPTION

This class provides a gadget similar to the palette.gadget and the GadTools palette gadget. Key difference is that this gadget provides more than just a "pick a color" user interface. Instead of supplying a pen index, this gadget requires color values. This gadget also provides tags that make it easy to change the colors of the palette and integrate this object with other gadgets using a model.

METHODS

OM\_NEW -- Create the palette gadget, calls OM\_SET. Passed to

---

superclass.

OM\_SET -- Set object attributes. Passed to superclass first.

OM\_GET -- Get object attributes.

OM\_NOTIFY -- Notify connections of attribute changes.

OM\_DISPOSE -- Frees resources and calls superclass.

OM\_UPDATE -- Calls OM\_SET.

GM\_DOMAIN -- Return the maximum and minimum size of the gadget.

GM\_RENDER -- Renders the gadget imagery. Overrides the superclass.

GM\_GOACTIVE -- Activate the gadget on mouse click only. Overrides the superclass.

GM\_HANDLEINPUT -- Handles input events once active. Overrides the superclass.

GM\_GOINACTIVE -- Deactivates the gadget. Passed to the superclass.

GM\_LAYOUT -- On initial layout, this object sends out OM\_NOTIFY

All other methods are passed to the superclass.

#### ATTRIBUTES

TCPALETTE\_ShowSelected (BOOL)

Keeps the selected pen highlighted when user input ends.

Defaults to TRUE

Applicability is (OM\_NEW, OM\_SET, OM\_GET)

TCPALETTE\_SelectedColor (UBYTE) [0..255]

Selected color

Defaults to 0

Applicability is (OM\_NEW, OM\_SET, OM\_UPDATE, OM\_GET, OM\_NOTIFY)

TCPALETTE\_SelectedRGB (struct TCPaletteRGB \*)

32bit per component, Red, Green & Blue value of the selected color

.

Applicability is (OM\_NEW, OM\_SET, OM\_UPDATE, OM\_GET, OM\_NOTIFY)

TCPALETTE\_SelectedLRGB (struct TCPaletteLRGB \*)

8bit per component Red, Green & Blue value of the selected color.  
This color format is packed into a ULONG in this format 0x00rrggbb

Applicability is (OM\_NEW, OM\_SET, OM\_UPDATE, OM\_GET, OM\_NOTIFY)

TCPALETTE\_Precision (ULONG) [1..32]

Bit depth of component colors, only effects the values of the following three tags. This was implemented because some gadgets cannot handle full 32bit numbers, specifically the slider.gadget.

Defaults to 8

Applicability is (OM\_NEW, OM\_SET, OM\_UPDATE, OM\_GET, OM\_NOTIFY)

TCPALETTE\_SelectedRed (ULONG)  
 TCPALETTE\_SelectedGreen (ULONG)  
 TCPALETTE\_SelectedBlue (ULONG)

Component level of the selected color. The value of these attributes is dependant on the TCPALETTE\_Precision attribute. Setting TCPALETTE\_Precision to 8, will yield a range from 0-255; 4 = 0-15

Applicability is (OM\_NEW, OM\_SET, OM\_UPDATE, OM\_GET, OM\_NOTIFY)

TCPALETTE\_NumColors (ULONG) [1..256]  
 Number of colors in the palette.

Defaults to 1

Applicability is (OM\_NEW, OM\_SET, OM\_UPDATE, OM\_GET, OM\_NOTIFY)

TCPALETTE\_RGBPalette (struct TCPaletteRGB \*)  
 Palette data, you provide an array, with at leaset TCPALETTE\_NumColors entries. If your array is to small, bad things will happen (especially with OM\_GET).

Defaults to all entries 0,0,0

Applicability is (OM\_NEW, OM\_SET, OM\_UPDATE, OM\_GET)

TCPALETTE\_LRGBPalette (struct TCPaletteLRGB \* or ULONG \*)  
 Palette data, you provide an array, with at leaset TCPALETTE\_NumColors entries. If your array is to small, bad things will happen (especially with OM\_GET).

Defaults to all entries 0

Applicability is (OM\_NEW, OM\_SET, OM\_UPDATE, OM\_GET)

TCPALETTE\_Orientation  
 Restrict layout to horizontal or vertical.  
 Not yet implemented.

TCPALETTE\_EditMode (ULONG)  
 TCPEM\_NORMAL -  
 TCPEM\_COPY - The color of the current selected pen will be copied to the pen the user selects next.  
 TCPEM\_SWAP - The color of the current selected pen will be swaped with the pen the user selects next.  
 TCPEN\_SPREAD - The color of the current selected pen will be spread to the pen the user selects next.

Applicability is (OM\_NEW, OM\_SET, OM\_UPDATE, OM\_GET, OM\_NOTIFY)

---

#### TCPALETTE\_Undo

Undo changes to palette, this should be a method.

Applicability is (OM\_SET, OM\_UPDATE)

#### TCPALETTE\_NoUndo

If zero the undo buffer has data in it.  
(so it's mappable to GA\_Disabled, ie. disable UNDO gadget)

Applicability is (OM\_GET, OM\_NOTIFY)

#### NOTIFICATION

These tags are supplied during notification.

TCPALETTE\_SelectedColor (UBYTE) [0..255]

TCPALETTE\_SelectedRGB (struct TCPaletteRGB \*)

TCPALETTE\_SelectedLRGB (struct TCPaletteRGB \*)

TCPALETTE\_SelectedRed (ULONG)

TCPALETTE\_SelectedGreen (ULONG)

TCPALETTE\_SelectedBlue (ULONG)

TCPALETTE\_NumColors (ULONG) [1..256]

TCPALETTE\_EditMode

TCPALETTE\_NoUndo (ULONG)

#### NOTES

On true color CyberGfx displays, this gadget renders true colors.  
On 8bit or less displays, this gadget uses FindColor() to display colors, which means that the gadget may become visually corrupt when other programs change the screen colors. So you may want to rerender the object periodically.

#### BUGS

Gadget may become visually corrupt when screen colors change on 8 bit or less screens.

#### SEE ALSO