

TMCFlashServer & TMCFlashLabel Components

The [TMCFlashServer](#) and [TMCFlashLabel](#) components in this package are Delphi VCL components. TMCFlashLabel cannot work without a TMCFlashServer, and TMCFlashServer has nothing to do without at least one active TMCFlashLabel. The TMCFlashLabel can flash in two different ways, either on/off or cycling through a set of colors.

[Ordering information](#)

[Contacting the author](#)

TMCFlashServer Component

See Also Property

The only purpose of this component is to supply one or more flashing labels with timer ticks. This is done to prevent eating up all available timers (maximum of 32 under Win 3.x). You only need one TMCFlashServer active at any one time, but if there isnt one available, no flashing will occur in the labels.

The OnlyActiveForm property determines whether all labels flash or only the ones on the currently active form.

TMCFlashLabel

OnlyActiveForm

OnlyActiveForm Property

Declaration:

property OnlyActiveForm: Boolean;

Description:

This determines whether all labels flash or only the ones on the currently active form. Since a [TMCFlashServer](#) component can service many forms full of [TMCFlashLabel](#) components, theres got to be a way to control which ones flash. If youve got two or more forms onscreen, but only one is active, the flashing on the other forms may distract the users attention.

TMCFlashLabel Component

Properties Events

This is the meat of the package, the component that adds something useful to your Delphi programs. It adds movement and color, and can get the users attention whenever something needs pointing out. Great for multimedia apps and presentations too!

The label can flash in any of two ways, either by blinking (visible/invisible) or by cycling through a set of colors. This is controlled by the value of FlashType, that can be either ftOnOff or ftColorCycle.

The frequency of the flashing is controlled by FlashInterval measured in 40ms pieces. This means that a value of 25 gives you an interval of exactly one second, while 12 gives approx a $\frac{1}{2}$ s econd interval.

When FlashType is ftColorCycle, the label cycles through the colors in FlashColors. Ive made life easy for you and provided a property editor for editing this, so you wont have to try to enter hex numbers or something like that.

Properties

- ▶ Run-time only
- 🔑 Key properties

<u>Align</u>	🔑 <u>FlashInterval</u>	<u>PopupMenu</u>
<u>Alignment</u>	🔑 <u>FlashType</u>	<u>ShowAccelChar</u>
<u>AutoSize</u>	<u>FocusControl</u>	<u>ShowHint</u>
<u>Caption</u>	<u>Font</u>	<u>Tag</u>
<u>Color</u>	<u>Height</u>	<u>Top</u>
<u>Cursor</u>	<u>Hint</u>	<u>Transparent</u>
<u>DragCursor</u>	<u>Left</u>	<u>Visible</u>
<u>DragMode</u>	<u>Name</u>	<u>Width</u>
<u>Enabled</u>	<u>ParentColor</u>	<u>Wordwrap</u>
🔑 <u>FlashColors</u>	<u>ParentFont</u>	<u>ParentShowHint</u>

Events

OnClick

OnDbClick

OnDragDrop

OnDragOver

OnEndDrag

OnMouseDown

OnMouseMove

OnMouseUp

FlashType Property

Declaration:

property FlashType: TMCFlashType;

Description:

The label can flash in any of two ways, either by blinking (visible/invisible) or by cycling through a set of colors. This is controlled by the value of FlashType, that can be either ftOnOff or ftColorCycle. If you select ftColorCycle, the colors are in [FlashColors](#).

FlashInterval Property

Declaration:

property FlashInterval: integer;

Description:

This is measured in 40ms intervals, setting a value of 1 gives you 25 flashes/second. The timing logic in TMCFlashServer uses a regular Delphi [TTimer](#), so the timing isnt all that precise - the precision is somewhere around 55ms. Heavy processor / disk load can also cause the ticks to come at irregular intervals, but this is normal for Windows applications.

FlashColors Property

Declaration

property FlashColors: TColorCycling;

Description:

This is the storage area for colors used when FlashType is set to ColorCycle. The TColorCycling type is internal to the FlashLabel unit, and you'll never have to use it - I've provided you with an easy to use property editor for this value, where you can see at a glance what your color cycling will look like.

Ordering information

The registration fee for the component is \$20 without source and \$40 with source. This is for electronic registration only, you have to supply us with an electronic address (CompuServe, InterNet or FidoNet) to which we can send the registered version. If you want a disk, contact us!

Whatever way you do it, you receive the same benefits - a version that works outside the IDE. In addition your conscience will stop hurting, and you'll be able to distribute programs using the component (unregistered version requires Delphi to be running).

You can register this pack in four ways:

CompuServe SWREG, register #10734 (without source, \$20) or #10735 (includes source, \$40)

Sending check or money order drawn on a Swedish bank

Sending cash in an envelope (if you dare)

Direct bank transfer, using the below information:

Account 8313-9 3.441.335-1

Kristianstads Sparbank

Via SwedBank, Stockholm

Swift Code SWEDSESS

Our mailing adress is as follows (see [Contacting the author](#) for other addresses)

MegaComm HB
Att Christian Tiberg
V Storgatan 67A
S-291 53 Kristianstad
SWEDEN

Contacting the author

We can be reached through a lot of methods and addresses:

CompuServe: 100777,2775

InterNet: ctiberg@silver.ct.se

FidoNet: 2:200/602

Phone: +46-44-103435

Fax: +46-44-248874

BBS: +46-44-125230

Snail mail:

MegaComm HB

Att Christian Tiberg

V Storgatan 67A

S-291 53 Kristianstad

SWEDEN

