

## **Mouse Blaster Ver 1.0**

MouseBlaster™ is a 32 bit, user configurable custom control created to give the developer power over how the mouse interacts with the end user. The MouseBlaster control provides options for controlling windows mouse messages on any active window. Developed by [Visual Enterprises](#), MouseBlaster™ was designed for Visual Basic 5 programmers. This custom control (OCX) is distributed as shareware and carry's a small registration fee to those who wish to incorporate MouseBlaster™ in their applications.

*Mouse Blaster™ was created by John T. Macklin and Visual Enterprises.  
Email: [jtmacklin@hotmail.com](mailto:jtmacklin@hotmail.com) ; [jtmackli@ktis.net](mailto:jtmackli@ktis.net)*

## [Topic Index](#)

Installing MouseBlaster™  
How to Register with Visual Enterprises  
Mouse Blaster Properties  
Mouse Blaster Functions

### [MouseBlast Properties](#)

Caption  
DblClickSpeed  
MsgAction  
MsgFound  
WinHwnd  
WinMsgTrap

## Installing Mouse Blaster Ver 1.0

MouseBlaster Ver 1.0 is designed for both Windows 95 and NT 4.0 Operating Systems. Mouse Blaster™ was developed and has been tested for developers using Visual Basic 5.0. There are no known implementation problems but we sincerely recommend that you follow the directions given in this help manual. Before you can use the MouseBlast.Ocx in your application, or the test application, you must register the file by running Regsvr32.EXE from your Windows 95/NT WinSys Directory.

Once this is done you may select Mouse Blaster from within the Components list of the Visual Basic 5.0 development environment. The Components list can be loaded by pressing CTL + T from the Visual Basic Project Design Window.

### Installation Procedures

1. Unpack the MouseBlast.Ocx from the MouseBlast.Zip Archive.
2. Copy the MouseBlast.Ocx to the WinSys Directory of your operating system. Windows 95 C:\Windows\System, Windows NT C:\Winnt\System32.
3. Register the control by double clicking on the MouseBlast.Ocx file.
4. From the **OpenWith** Window, Uncheck the **always use this program to open this file** and then click on the **Other** command button.
5. In the File Name text box click on the Regsvr32.exe and choose ok.
6. You will then see the message box " Registration of MouseBlast.Ocx succeeded ".
7. You are now ready to add the control to your project from the Visual Basic component list by pressing CTL+T.

### How to Register Mouse Blaster

You may register Mouse Blaster™ by Printing and completing the Registration Form included in this manual. Include the reference product serial number 11-1564. You may also register by email by using the Email Registration form, which was distributed with MouseBlast.Zip. There is a one-time registration fee of \$10.00 US Currency. By registering Mouse Blaster you will receive future updates via email and also notification of future development tools. We hope you enjoy the power of Mouse Blaster...

MouseBlas Help was Written By John T. Macklin.

This product is Owned by Visual Enterprises 1998 All Rights Reserved .

Visual Enterprises 1059 Ravenwood , Montgomery City MO 63361

Email [jtmacklin@hotmail.com](mailto:jtmacklin@hotmail.com) , [jtmackli@ktis.net](mailto:jtmackli@ktis.net)

### Mouse Blaster Properties

**Caption** " String for MouseBlast Control Identification "

**DblClickSpeed** " Mouse Double Click Speed in milliseconds "

**MsgAction** " Determines if message is trapped or passed "

**MsgFound** " Indicates a message was processed "

**WinHwnd** " The handle of target window "

**WinMsgTrap** " The Mouse Message to Filter from the Queue "

## Caption Property

**Caption** " String for MouseBlast Control Identification "

Returns or sets the Mouse Blaster control's Caption .

Syntax

object.Caption [= "String"]

The Caption property syntax has these parts:

<u>Part</u>	<u>Description</u>
-------------	--------------------

object	- An object expression that applies to the MouseBlast Control.
--------	--

Caption	- A string expression specifying the MouseBlast Control's Caption.
---------	--

## DbClickSpeed Property

**DbClickSpeed** " Mouse Double Click Speed in milliseconds "

Returns or sets the windows mouse Double Click Speed .

Syntax

object.DbClickSpeed[= integer]

The DbClickSpeed property syntax has these parts:

<u>Part</u>	<u>Description</u>
-------------	--------------------

object	- An object expression that applies to the MouseBlast Control.
--------	--

DbClickSpeed	- An integer value 0 - 999 specifying the windows double click speed in milliseconds.
--------------	---

## MsgAction Property

**MsgAction** " Determines if message is trapped or passed "

Returns or sets the MouseBlast control's message queue action .

Syntax

object.MsgAction[= MsgNoRemove] or [=MsgRemove]

The MsgAction property syntax has these parts:

<u>Part</u>	<u>Description</u>
-------------	--------------------

object	- An object expression that applies to the MouseBlast Control.
--------	--

MsgAction	- A constant specifying the controls message queue action.
-----------	--

### **Remarks:**

By setting the constant to MsgRemove or 1 the control will remove the message specified by the WinMsgTrap property from the message queue. Target window queue is identified by the WinHwnd property. If the constant setting is set to MsgNoRemove or 0 the control will not remove the specified message from the target window message queue.

## MsgFound Property

**MsgFound** " Indicates a message was processed "

Returns the MouseBlast control's MsgFound property.

Syntax

Msg = object.MsgFound    Read Only

The MsgFound property syntax has these parts:

<u>Part</u>	<u>Description</u>
-------------	--------------------

object	- An object expression that applies to the MouseBlast Control.
--------	--

MsgFound	- A integer value specifying the control filtered a message.
----------	--

Returns 1 or 0.

### **Remarks:**

When an installed filter which meets the developers filter criteria encounters the message to be filtered as specified by WinMsgTrap property the MsgFound property returns 1. When no messages were filtered the MsgFound property returns 0. The MsgFound value is not latched and returns 0 until a message is filtered then resets. Returns 1 only while the message is processed by the control.

## WinHwnd Property

**WinHwnd** " The handle of target window "

Return or sets the MouseBlast control's target window handle.

Syntax

object.WinHwnd [= long ] or [=Form1.hWnd]

The WinHwnd property syntax has these parts:

<u>Part</u>	<u>Description</u>
-------------	--------------------

object	- An object expression that applies to the MouseBlast Control.
--------	--

WinHwnd	- A long value specifying the handle of the target window whose message queue is to be filtered .
---------	---

### **Remarks:**

The WinHwnd property designates the target window in which the message queue filter is to be installed. You can obtain this value by returning the value from Form1.hWnd or from a window API function such as FindWindowEx.

## WinMsgTrap Property

**WinMsgTrap** " The Mouse Message to Filter from the Queue "

Return or sets the MouseBlast control's mouse message to be filtered.

Syntax

object.WinMsgTrap [= WM\_RBUTTONDOWN ] or [=516]

The WinMsgTrap property syntax has these parts:

Part    Description

object - An object expression that applies to the MouseBlast Control.

WinMsgTrap - A constant expression or value specifying the windows mouse message to be removed from the queue.

### Remarks:

The WinMsgTrap property contains these defined constants and their associated return values.

WM_MOUSEMOVE	Returns	512
WM_LBUTTONDOWN	Returns	513
WM_LBUTTONUP	Returns	514
WM_LBUTTONDOWNBLCLK	Returns	515
WM_RBUTTONDOWN	Returns	516
WM_RBUTTONUP	Returns	517
WM_RBUTTONDOWNBLCLK	Returns	518
WM_MBUTTONDOWN	Returns	519
WM_MBUTTONUP	Returns	520
WM_MBUTTONDOWNBLCLK	Returns	521

## FilterMsg and RemoveFilter Functions

### FilterMsg Function

The FilterMsg Function install's the developer defined filter and should be called after setting the MessageBlast control's MsgAction, WinHwnd and WinMsgTrap properties. This function will then call the address of the target windows message queue and install the filter specified by the WinMsgTrap property.

#### Syntax

Object.FilterMsg

### RemoveFilter Function

The RemoveFilter Function is used to restore the old message queue and remove any developer defined message filters. It is recommended to call this function in your cmdExit\_Click event or where your application's unload code is executed. If this is not properly implemented in code you can cause applications running in windows to crash unexpectedly.

#### Syntax

Object.RemoveFilter

Note: Object refers to the instance of the MessageBlast control.

## Licensing Information

Terms by which we (Visual Enterprises 1059 Ravenwood Montgomery City, MO 63361 USA ) permit other parties to use or distribute the Shareware Control "Mouse Blaster".

**BY COPYING, USING OR DISTRIBUTING THIS SHAREWARE , YOU INDICATE YOUR AGREEMENT TO THE LICENSING TERMS AND CONDITIONS.**

*"Mouse Blaster" means the Shareware version of "MouseBlst.OCX" and its related files ( including this help file ) as owned by "Visual Enterprises"*

### **ATTENTION ALL SHAREWARE USERS**

Everyone can and is encouraged to copy, upload and generally pass around "Mouse Blaster" without charging for it. However YOU MAY NOT delete, edit or alter any of its files including this help document. If you continue to use Mouse Blaster or include it in your programs you are expected to register it with Visual Enterprises by sending in your \$10.00 registration fee.

### **ATTENTION ALL SHAREWARE DISTRIBUTORS**

Visual Enterprises owner and copyright holder of "Mouse Blaster" allow and encourage most types of distribution, provided that our licensing terms are observed. Your right to distribute "Mouse Blaster" is personal, and does not include the right to sublicense, lease or rent it as part of any hardware or software package without prior permission.

*Distribution of this software in violation of the licensing terms and conditions , is prohibited and subject to penalty by law.*

## Licensing Agreement Terms and Conditions

- [1] YOU MAY NOT MODIFY, DELETE OR RENAME ANY PROGRAM FILE.
- [2] YOU MAY NOT CHARGE MORE THAN U.S. \$ 10 FOR THIS CONTROL.
- [3] YOU MUST MAKE IT CLEAR TO YOUR CUSTOMERS THAT SHAREWARE IS NOT FREE SOFTWARE, BUT A REGISTRATION FEE IS EXPECTED IF THE PROGRAM IS TO BE USED ON A REGULAR BASIS.
- [4] THE NAME "Visual Enterprises" MUST BE MENTIONED IN ANY CATALOG OR ON-LINE DESCRIPTION OR PACKAGING ASSOCIATED WITH THIS PROGRAM.
- [5] YOU MUST TRY TO SELL ONLY THE MOST CURRENT VERSION.

## Registering Mouse Blaster Ver 1.0 Shareware

**To register your copy of Mouse Blaster Ver 1.0 Fill out and Print the form below and mail or Email the completed Register.txt file included with the MouseBlst.zip file.**

Business: \_\_\_\_\_

Name: \_\_\_\_\_ Address: \_\_\_\_\_

Telephone# \_\_\_\_\_ Fax# \_\_\_\_\_ [ ] Other \_\_\_\_\_

Company URL : \_\_\_\_\_

Your Name: \_\_\_\_\_ Title: \_\_\_\_\_

Product Name: \_\_\_\_\_

\$ 10 enclosed to cover the cost of license, as well as future controls and updates. Check, Money Order and Cash are accepted.

*Your one time Registration Fee gives you rights to use Mouse Blaster and any future controls developed as shareware by Visual Enterprises .*

**Visual Enterprises**  
**1059 Ravenwood**  
**Montgomery City , MO 63361 USA**

**Voice: +1 - 573-564-7914**  
**FAX: +1 - 573-564-7914**  
**Internet: [jtmacklin@hotmail.com](mailto:jtmacklin@hotmail.com)**

