

CASL s o f t VB Error Ease 2.0 Custom Control

Online Programmers Guide

[Introduction](#)

[License Agreement](#)

[Royalty Free Distribution](#)

[VB Error Ease Overview](#)

[VB Error Ease Properties](#)

[VB Error Ease Events](#)

[The Histroy Log File](#)

[Frequently Asked Questions](#)

[Contacting CASL s o f t](#)

Introduction

As many VB programmers know, or will eventually learn, writing an efficient maintainable error handler for VB applications poses quite a challenge. Any of us who have turned a simple On Error Goto routine into a nightmare of If/Then/Elses, Resumes, MsgBoxs, and what ever else came up after design, can breathe a sigh of relief. VB Error Ease provides a much simpler approach to error processing. For any of those lucky ones who have not had to deal with this BASIC initiation, enjoy your good fortune and don't look back. VB Error Ease will shelter you from this and leave you wondering what all the fuss was about.

How often have you wished you knew the sequence of events leading to an application error or problem situation? VB Error Ease provides the application programmer with an easy way to create application run time history logs. Placing these entries at given checkpoints throughout your application can prove valuable and time saving if your application users run into software problems. You simply look at the text log file or files to determine where the problem occurred and what the order of events were that led to the problem.

VB Error Ease is an enhanced error handling tool originally designed for Visual Basic. However, because it is an OLE custom control, it may be used in any development environment that supports OCX's. Therefore, you can utilize the power of VB Error Ease in all of your development projects.

When added to your project, VB Error Ease will provide the VB programmer with a number of properties and events specifically dedicated to the handling and processing of application errors and message logging. These new resources will enable you to quickly create efficient error handling routines as well as useful informative activity logs for your applications. VB Error Ease can be used simply for logging messages to an activity history log file or as an all encompassing error handling process complete with custom error tables, error properties, and error trapping events. VB Error Ease will considerably reduce the amount of VB error handling code needed in your VB applications. In addition, VB Error Ease provides a common error handling mechanism that will efficiently handle multiple applications concurrently. This in turn yields significantly less memory usage for your application users.

VB Error Ease takes a refreshing approach to exception handling. The exceptions, or messages, are maintained in a data file or "Message Table". This approach enables VB Error Ease to be completely data driven. In turn, you the developer can make changes to this table without impacting, retesting or recompiling your source code. In addition, in the event that an error comes up that you have not specifically coded for at design time, you can simply add the error number and appropriate message to the "Message Table" and the error is handled from that point forward. No more recompiles!

CASL s o f t has included some powerful tools to help you take advantage of the benefits of VB Error Ease. The first is the CS Message Table Facility. This tool is designed to help you quickly and easily create and maintain your VB Error Ease message table(s). In addition, the VBEASE Code Generator application has also been included. This tool is designed to enable you to quickly add VB Error Ease message handling to your VB applications with a few clicks of the mouse. VBEASE turns the tedious manual ritual of VB error coding into an automated task. The generated code is designed to run seamlessly with your code and the VB Error Ease OCX.

CASL s o f t License Agreement

CASL s o f t License Agreement

This is a legal agreement between you the end user, and CASL s o f t. By stating your intention to agree to the terms of this license agreement during the installation process of this software package, you are agreeing to be bound by the terms of this agreement. If you do not agree to the terms of this agreement, promptly return this un-installed software diskette and the accompanying items(including written documentation) to the place you obtained them, for a full refund.

1. GRANT OF LICENSE. CASL s o f t grants you the right to use one copy of the enclosed CASL s o f t software program(s) (the "SOFTWARE") on a single terminal connected to a single computer (i.e., with a single CPU). You may not network the SOFTWARE or otherwise use it on more than one computer or computer terminal at the same time.

2. COPYRIGHT. The SOFTWARE is owned by CASL s o f t and is protected by United States copyright laws and international treaty provisions. Therefore, you must treat the SOFTWARE like any other copyrighted material (e.g, a book or musical recording). except that you may either (a) make one copy of the SOFTWARE solely for backup or archival purposes, or (b) transfer the SOFTWARE to a single hard disk provided you keep the original solely for backup or archival purposes. You may not copy the written materials accompanying the SOFTWARE.

3. OTHER RESTRICTIONS. You may not rent or lease the SOFTWARE, but you may transfer the SOFTWARE and accompanying written materials on a permanent basis provided you retain no copies and the recipient agrees to the terms of this agreement. You may not reverse engineer, decompile or disassemble the SOFTWARE. If SOFTWARE is an update, any transfer must include the update and all prior versions.

4. RUNTIME AND DESIGNTIME SOFTWARE. If the SOFTWARE is a Custom Control, and both runtime ("RUNTIME) and designtime ("DESIGNTIME") versions are included in the SOFTWARE package, then the DESIGNTIME version is subject to the restrictions set forth above and may not be distributed with your applications. However, CASL s o f t grants you a royalty-free right to reproduce and distribute the RUNTIME version of the SOFTWARE with you Visual Basic applications, provided that (a) you distribute the RUNTIME version only in conjunction with and as part of your applications; (b) you do not use CASL s o f t's name, logo or trademarks to market you applications without express written consent of CASL s o f t; (c) you include CASL s o f t's copyright notice for the SOFTWARE on your product label; and (d) you agree to indemnify, hold harmless and defend CASL s o f t from and against any claims or lawsuits, including attorney's fees, that arise or result from the use or distribution of your applications with the RUNTIME modules. The RUNTIME modules are limited to the runtime OCX file(s).

Limited Warranty

LIMITED WARRANTY. CASL s o f t warrants that the original disks are free from defects for 90 days from the date of delivery of the SOFTWARE.

CUSTOMER REMEDIES. CASL s o f t's entire liability and your exclusive remedy shall be, at CASL s o f t's option, either (a) return of the price paid or (b) replacement of the SOFTWARE which does not meet CASL s o f t's Limited Warranty and which is returned to CASL s o f t with a copy of your receipt. Any replacement SOFTWARE will be warranted for the remainder of the original warranty period or 30 days, whichever is longer. THESE REMEDIES ARE NOT AVAILABLE OUTSIDE OF THE UNITED STATES OF AMERICA.

NO OTHER WARRANTIES. CASL s o f t disclaims all other warranties, either express or implied, including but not limited to implied warranties of merchantability and fitness for a particular purpose, with respect to the SOFTWARE and the accompanying written materials. This limited warranty

gives you specific legal rights. You may have others which vary from state to state.

NO LIABILITY FOR CONSEQUENTIAL DAMAGES. In no event shall CASL s o f t or its suppliers be liable for any damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or other pecuniary loss) arising out of use or inability to use the SOFTWARE, even if CASL s o f t has been advised of the possibility of such damages. Because some states do not allow the exclusion of limitation of liability, for consequential or incidental damages, the above limitation may not apply to you.

This agreement is governed by the laws of the State of Connecticut.

Royalty Free Distribution

CASL s o f t grants you, the registered user of VB Error Ease 2.0, the right to distribute the CSerr.OCX file with your completed applications. This is meant as a run time only OCX. All terms of the software License Agreement are contained in the CASL s o f t license agreement.

VB Error Ease Overview

As stated in the introduction, VB Error Ease is an enhanced error handling tool. This tool has been designed to provide the programmer with a comprehensive process by which application messages can be captured and handled. Throughout this manual the word message will often be used in conjunction or in place of the word error. This has been done for a specific reason. The reason being that the VB Error Ease tool is not solely intended to process just application errors. While a large part of professional software development is dedicated to the handling of application errors, a significant portion of time is spent keeping the application user aware of ongoing events pertaining to the task at hand. This is where VB Error Ease goes beyond error handling and into message handling.

Implementation Example 1:

For example: A payroll program is in the process of calculating a commission for a given salesperson and the amount of the calculation exceeds \$3000.00. If the maximum commission is \$3000.00, the application determines that the user should be notified. At this point the VB Error Ease message handling process would be invoked by setting the ErrorNumber property to 12345. This in turn will cause VB Error Ease to open the message table that has been assigned to the ErrorTableName property. VB Error Ease will search through this table to find a matching entry for message number 12345 and display the appropriate message to the user.

As with all Windows MessageBoxes, at least one button choice is presented to the user for response. VB Error Ease determines what choice(s) the user is given based on the Message Level that has been set up for this particular message. See the "Message Table Facility Help", for more information. In addition to presenting this important piece of information to the user, VB Error Ease will also write all information pertaining to this message to an output history log file. The history log file is a text file that can be looked at with a text editor such as the Windows Write word processor. The name and location of the output log file are set using the ErrorLogName property. This history log file can later be used to review what has happened during the course of the application execution. This becomes very useful when tracking possible oversights as well as possible application problems.

Implementation Example 2:

The following is another example of how VB Error Ease can be applied to an application.

VB Error Ease can be used to handle all generic user interaction for an entire application. Suppose the payroll application referenced in the first example was distributed to several countries. By using VB Error Ease to handle application messages, no coding modifications would be required. Instead, the message table provided with the specific international version of the application would have all of the same messages but written in its respective language. This could substantially reduce the amount of coding required to provide application messaging in multiple languages.

These simple examples show some of the basic flows of VB Error Ease as well as some useful applications. However, in other help topics, an in depth look at all of the components of VB Error Ease will demonstrate how complex message handling can be easily added to your applications. This is done while allowing the customization required for the individual application. As the term "Custom Control" denotes, VB Error Ease can truly be used to specifically tailor the application message handling stream to fit a unique purpose.

VB Error Ease Properties

Just as with all OCX controls, VB Error Ease provides the programmer with a set of custom properties. This set of properties is used to control the behavior of the VB Error Ease control. It will become evident how the combination of different VB Error Ease properties and events will enable the programmer to quickly add efficient, customized error and or message handling to applications.

The following is a comprehensive look at each of the VB Error Ease custom properties:

[Name](#)

[Index](#)

[Left](#)

[Top](#)

[Tag](#)

[Enabled](#)

[ErrorNumber](#)

[DefaultErrorNumber](#)

[ErrorLine](#)

[ErrorTableName](#)

[ErrorLogName](#)

[ErrorLogText](#)

[ErrorBoxResponse](#)

[ErrorBoxTitle](#)

[ErrorProcedure](#)

[ErrorMessageVB](#)

[ErrorEXEName](#)

[ErrorUserID](#)

[ErrorModule](#)

[ErrorLevel](#)

[ErrorLevelOVD](#)

[ErrorResult](#)

[ErrorMemoryEnabled](#)

[LogOnly](#)

[MessageLiteral](#)

[LogEnabled](#)

[BypassLog](#)

[TableEnabled](#)

[LogFormat](#)

[UserLogField1 - UserLogField5](#)

[UserLogTitle1 - UserLogTitle5](#)

[DisplayNumber](#)

[DeveloperMode](#)

Name

Data Type: Text

Property Description:

The Name property specifies the name used in code to identify the VB Error Ease control.

Index

Data Type: Integer

Property Description:

The Index property specifies the control index used in code to identify the VB Error Ease control when used in a control array.

Left

Data Type: Integer

Property Description:

The Left property determines the distance between the internal left edge of an object and the left edge of its container.

Top

Data Type: Integer

Property Description:

The Top property determines the distance between the internal top edge of an object and the top edge of its container.

Tag

Data Type: Expression

Property Description:

The Tag property stores any unique data needed by the program. The value of the Tag property is not used by VB Error Ease for any purpose. This property can be used at the programmer's discretion for a custom data property.

Enabled

Data Type: Boolean

Property Description:

The Enabled property determines whether the VB Error Ease control can respond to application-generated events.

ErrorNumber

Data Type: Integer

Property Description:

The ErrorNumber property is used to pass the current error or message number to the VB Error Ease message handling routines. This property is used by VB Error Ease as the "trigger" for firing the VB Error Ease message handling routines. As a standard, the ErrorNumber property should be the last property that a particular sub routine or function should set when invoking VB Error Ease. This property is the VB Error Ease default property.

DefaultErrorNumber

Data Type: Integer

Property Description:

The DefaultErrorNumber property is used to set a default value in the event that the number passed to the ErrorNumber property is not found in the Message Table. This property is useful for providing a "catch all" error message to the application user. The message must be entered in the Message Table as all other VB Error Ease messages. However, the true message number will be logged to the history log file to provide the developer with the actual error information.

ErrorLine

Data Type: Long

Property Description:

The ErrorLine property is used to pass the current error line to the VB Error Ease message handling routines. This is retrieved in VB by using the VB function Erl. This property is specifically used as pass-along information for the history log file.

ErrorTableName

Data Type: Text

Property Description:

The ErrorTableName property is used by VB Error Ease to determine the name and location of the desired message table file to access. This property is generally set up once at run time and is unchanged for the duration of the application. However, this property can be changed as often as seen fit in order to access unique message tables to accommodate unique situations. It is useful to use the application INI file to store the path and file name for the ErrorTableName. This allows for moving of this file without the need to recompile code.

ErrorLogName

Data Type: Text

Property Description:

The ErrorLogName property is used by VB Error Ease to determine the name and location of the desired history log output file. This property is generally set up one time and is unchanged for the duration of the application. However, this property can be changed as often as seen fit in order to update unique history logs. This is a useful feature when creating history logs for different types of messages. A Separate log may be desired for communication messages while another log may be desired for input validation errors. It is useful to use the application INI file to store the path and file name for the ErrorLogName. This allows for moving of this file without the need to recompile code.

ErrorLogText

Data Type: Text

Property Description:

The ErrorLogText property is used to pass free form text from the application to VB Error Ease. This property is specifically used as pass along information for the history log file.

ErrorBoxResponse

Data Type: Integer

Accessibility: Not available at design time.

Property Description:

The ErrorBoxResponse property is used to retrieve the value of the message box user choice. These values correspond to the values returned by the Windows MessageBox functions. This property can be used in conjunction with the ErrorResult to determine where control is passed following an VB Error Ease invoked message. Typically, the ErrorResult is sufficient for determining program flow following a message. However, to provide additional flexibility to the developer, the ErrorBoxResponse will provide the value of the actual command button choice chosen by the application user.

ErrorBoxTitle

Data Type: Text

Property Description:

The ErrorBoxTitle property is used by VB Error Ease as the title that will appear in the MessageBox presented to the user. This property is generally set up one time and is unchanged for the duration of the application. However, this property can be changed as often as seen fit in order to accommodate unique situations.

ErrorProcedure

Data Type: Text

Property Description:

The ErrorProcedure property is used to pass the name of the procedure that caused VB Error Ease to be invoked. This property is specifically used as pass along information for the history log file. This information can be very useful when attempting to resolve an application problem. This is typically set at the beginning of a procedure that contains VB Error Ease message handling. VBEASE code generator will insert this code into a VB application if requested.

ErrorMessageVB

Data Type: Text

Property Description:

The ErrorMessageVB property is used to pass the current error description to the Error Ease message handling routines. This is retrieved, in Visual Basic, by using the VB function Error\$. This property is used by VB Error Ease in the event that a match for the ErrorNumber is not found in the message table file. This happens when a true VB error such as "Illegal Function Call" occurs. VB Error Ease will then have adequate information to present to the application user. If this property has not been set and an entry in the message table is not found, VB Error Ease will display a default message to the user. This property is typically loaded in the ErrorLogSet event of the CSerr.OCX control. See the CSSAMPLE application provided with this package.

ErrorEXENAME

Data Type: Text

Property Description:

The ErrorEXENAME property is used by VB Error Ease to store the executable filename of the application that is utilizing the Error Ease message handling routines.

ErrorUserID

Data Type: Text

Property Description:

The ErrorUserID property is used to pass a User Identification of the person using the application. This property is generally setup one time at the start of the application and is unchanged for the duration of the application. This property is specifically used as pass along information for the history log file. This information can be very useful when implementing an application on a Local Area Network.

ErrorModule

Data Type: Text

Property Description:

The ErrorModule property is used to pass the name of the form or module that contains the VB Error Ease invoked message. This property is specifically used as pass along information for the history log file. This information can be very useful when attempting to resolve an application problem. This is typically set at the beginning of a procedure that contains VB Error Ease message handling. VBEASE code generator will insert this code into a Visual Basic application if requested.

ErrorLevel

Data Type: Integer

Property Description:

The ErrorLevel property is used to retrieve the message severity level for the last VB Error Ease handled message. This property is specifically used as information for the history log. The Message Level can be overridden using the ErrorLevelOVD property or by resetting the ErrorLevel parameter to 0 or 60-99 in the ErrorTrap event. The ErrorLevel is retrieved from the current message table entry that corresponds with the ErrorNumber property that has been set. A default level of 40 is given to any message not found in the Message Table or overridden using the ErrorLevelOVD property. The following is a list of the possible Error Levels.

Message Level Range	Message Type	User Choice(s)
0	Log Only	n/a
1 - 10	Informational	OK
11 -20	Warning	YES/NO
21 - 30	Severe Warning	ABORT/RETRY/ IGNORE
31 - 40	Application Fatal	OK
60 - 99	Explicit Result	n/a

The ErrorLevel controls how the Message Box will be presented to the user. In addition, the ErrorLevel governs the value returned to the application in the ErrorResult property. See ErrorResult property for an explanation of how the "Upper" and "Lower" ranges of the Message Level are used.

A Word About Explicit Result Messages

The Explicit Result Message was created to be used as a programmatic response for a given error or message. It is used by the programmer to define a response to a given situation without the necessity for user intervention. For example, an application needs to trap VB error 7 ('out of memory') to determine whether or not to write its data to disk or leave it in memory. If the programmer knows that this is okay for this application, the programmer can simply make an entry in the Message Table for Error Number 7 and set the Message Level to any number between 60 and 99. The programmer can then add a line of code in his scope level error block that simply reads if CSerr1.Result = 60 perform the MemoryToDisk function and Resume Next. This is a simple example of how VB Error Ease offers many different programming options based on the specific needs of the application.

The Explicit Result Messages are not shown to the user. These messages by nature are messages that the programmer knows the necessary response to. Unlike standard VB Error Ease handled messages, Explicit Result Messages return the actual Message Level that has been determined for this message in the ErrorResult property.

ErrorLevelOVD

Data Type: Integer

Property Description:

The ErrorLevelOVD property is used to override the Message Level of a given VB Error Ease handled message. This property becomes very useful when an exception situation arises and a message needs to be presented differently than normal. For example, if an application is launching a secondary application and that application's executable file does not exist, the original application could terminate because of the VB error 'file not found'. However, if the application does not need to terminate because of this situation, you can add a comparison in the error block of that procedure for the VB error 53 and change the Error Level to an 11. This will cause the VB Error Ease handler to prompt the user to continue or exit the application. Assuming the user chooses to continue and the error block calls Resume Next for any continued warnings messages, the application will continue processing. This basic example demonstrates how Error Ease can be used flexibly to accommodate unique situations. The Message Level can also be overridden by resetting the ErrorLevel parameter to 0, or 60-99 in the ErrorTrap event. With 0 indicating a log only entry and 60-99 indicating an "Explicit Result Message".

By interrogating the ErrorNumber and ErrorProcedure in the ErrorLogSet event, a centralized message override mechanism can be created for the application. This means that if ErrorNumber 53(VB 'file not found') is to only be overridden for procedure 'Load_Text_File', it can be done as follows in the ErrorLogSet event. Otherwise ErrorNumber 53 will be handled as governed by its Message table entry or the lack of a message table entry.

```
Sub ErrorLogSet()  
    'Special Case Overrides.  
    Select Case CSerr1.ErrorNumber  
        Case 53  
            If CSerr1.ErrorProcedure = "Load_Text_File" then  
                CSerr1.ErrorLevelOVD = 1  
            End If  
        End Select  
    End Sub
```

ErrorResult

Data Type: Integer

Property Description:

The ErrorResult property is used to determine where program control is going to be passed following a VB Error Ease handled message. The ErrorResult is calculated by adding either 10 or 20 to the ErrorBoxResponse property value. 10 is added to the ErrorBoxResponse if the ErrorLevel for this message is in the lower 5 values of its given ErrorLevel range. 20 is added to the ErrorBoxResponse if the ErrorLevel for this message is in the upper 5 values of its given ErrorLevel range. This is done to allow the programmer a greater range of flexibility when handling "like" errors. The programmer is effectively given 2 different actions to take for each ErrorLevel within a given procedure. For example, assume that an application creates database records for order entry. Also assume that the order being taken is to be delivered to the customer and the customer address is blank. These messages are represented in the Message Table as error numbers 12345 and 12346 with message levels 1 and 10. In cases where the customer is picking up the merchandise, ErrorNumber 12345, the application would simply present the user with a Message Box indicating a blank address and have the error block pass control back to the main program. But in the case of the customer having merchandise delivered, ErrorNumber 12346, the application would present the user with the same Message Box indicating a blank address but then have the error block pass control to the GetNameAddress function to force the user to fill in this information. This property becomes very useful when an exception situation arises and the same message response needs to be handled differently based on the current process. To summarize, recall that the ErrorLevel controls the allowable response(s) from the application user, by using the upper and lower boundaries of the ErrorLevel you are able to add dual functionality for that given ErrorLevel within the same error scope.

ErrorMemoryEnabled

Data Type: Boolean

Property Description:

The ErrorMemoryEnabled property determines whether the applicable VB Error Ease message table is loaded into memory from disk or accessed directly from disk. By utilizing the Memory feature the access time for the message table and display of the Message Box is faster. However, the drawback to this is that large Message Tables can take up significant blocks of memory. In addition, if Message Tables are being changed frequently throughout the application, the latest one accessed is the one residing in memory. In general the property is not used. However, if response is critical and the memory is available it can be an effective way to enhance performance.

LogOnly

Data Type: Boolean

Property Description:

The LogOnly property is used to send a log entry directly to the history log file bypassing the application user. This property is useful when processing messages that have no immediate impact on the functionality of the application but are useful for later research. A good application of this is to log a data capture entry to the history log file to capture all incoming and outgoing communications traffic. By setting the data capture flag in the application on and utilizing VB Error Ease, the application can send pertinent information directly to the history log for later analysis. The LogOnly flag must be set each time it is to be used because this property, much like the ErrorNumber property, acts as a "trigger" to start the VB Error Ease process. It is reset by VB Error Ease. When the LogOnly property is set to TRUE, the message table is not accessed. This is strictly a message passed through VB Error Ease to the history log routines. To change a message that has been initiated using the ErrorNumber property to a Log Only message, refer to the ErrorTrap event.

MessageLiteral

Data Type: Boolean

Property Description:

The MessageLiteral property is used to determine whether VB Error Ease will append the corresponding Message literal to the message prior to presenting it to the application user. The following is a list of Message literals that correspond to the given ErrorLevel for the message.

Message Levels 1-10:

This is an application Informational message:
This application will continue as normal.

Message Levels 11-20:

This is an application warning message:
Do you wish to continue with this application?

Message Levels 21-30:

This is an application user message:
Choose Ignore to continue, Retry to try process
again, or Abort to terminate this application.

Message Levels 31-40:

This is an uncorrectable application message:
This application will terminate!

LogEnabled

Data Type: Boolean

Property Description:

The LogEnabled property determines if the VB Error Ease control will log entries to the history log file.

BypassLog

Data Type: Boolean

Property Description:

The BypassLog property informs VB Error Ease that the History log entry for this particular message should not be output. This property is reset to False after each VB Error Ease handled message.

TableEnabled

Data Type: Boolean

Property Description:

The TableEnabled property determines whether the VB Error Ease control will check the Message Table when processing a message. If this is set to False, Error Ease will respond as if the message is a standard Visual Basic error message.

LogFormat

Data Type: Integer(Enumerated)

Property Description:

The LogFormat property determines how the history log file output records will be written to disk. 0 - Text signifies that the log entries will be written in standard ASCII text format. This format includes log field titles. Logs created using this format can easily be read using any type of text editor capable of reading ASCII text file formats. The second format available is 1 - Comma Delimited. This format is used when creating log entries which will be imported into a secondary database application. This feature is very useful for creating customized reports based on the history log files.

*A Word of Caution: Be careful when dynamically switching the log format. VB Error Ease OCX writes the log entry to the file specified using the ErrorLogName property. In addition, VB Error Ease simply appends the new entry to the end of the history log file. If switching between file formats, it is recommended that separate log files are used to control which type of output that is required.

The following is a list of the fields within the comma delimited log entry along with its corresponding database data type:

Field Name	Database Type
Date/Time	Text
UserID	Text
Executable	Text
Module	Text
Procedure	Text
Message Number	Number
Message Level	Number
Line Number	Number
Message	Memo
User Response	Text
Log Text	Memo
User Field1	Text
User Field2	Text
User Field3	Text
User Field4	Text
User Field5	Text

Any fields specified as "Text" in the above table will be limited to 255 characters prior to being output to the history log file when using the 1 - Comma Delimited option.

UserLogField1 UserLogField5

Data Type: Text

Property Description:

The UserLogField1 - UserLogField5 properties are a set of 5 free-form text fields. These fields are useful for additional log information that may be needed. For example, there may be a need to log a mainframe screen name that was being accessed at the time an error occurred. This can be done by using one of the available UserLogFields to pass this information to the history log file. Also see the UserLogTitle1 - UserLogTitle5 properties.

UserLogTitle1 UserLogTitle5

Data Type: Text

Property Description:

The UserLogTitle1 - UserLogTitle5 properties are a set of 5 free-form titles that correspond to the UserLogField1 - UserLogField5 properties. These titles are used as headers when creating the history log file entries. These properties are specifically used as pass along information for the history log files.

DisplayNumber

Data Type: Boolean

Property Description:

The DisplayNumber property determines whether the VB Error Ease control will include the Message Number in the message box that is presented to the user.

DeveloperMode

Data Type: Boolean

Property Description:

The DeveloperMode property determines whether the VB Error Ease control will include debugging information in the message box that is presented to the user. This is very useful during development time when it is inconvenient to repeatedly refer to the history log file. Debugging information includes: Message Number, Line Number, Module Name, Procedure Name.

VB Error Ease Events

VB Error Ease provides the programmer with its own custom events. These events are actions triggered by the VB Error Ease control. It will become evident how the combination of different VB Error Ease properties and events will enable the programmer to quickly add efficient customized error and or message handling to your applications.

The following is a comprehensive look at each of the VB Error Ease custom events:

[ErrorTrap](#)

[ErrorLogSet](#)

ErrorTrap

The purpose of the ErrorTrap event is to give the programmer greater flexibility when designing the application message handling process. The programmer is given the ErrorTrap event as a "stopping" point prior to VB Error Ease presenting the Message to the application user. In affect this is a convenient place to add any special functionality to the application Message Handling process that may be specific to a given situation. For example, suppose that an application provides an option specifying whether or not to log message information to a VB Error Ease history log file. While in the ErrorTrap event the application can determine if the "Log flag" has been set to FALSE. If so, the ErrorAction parameter that has been provided by Error Ease can be set to the value 3. This will inform VB Error Ease to present the message to the application user but not to log the message information to the history log file.

Event Parameters:

ErrorNumber: The ErrorNumber parameter is an integer type. This is the Message Number that initiated the VB Error Ease process. Modifying this parameter has no effect on the processing of the message. This parameter is provided in the event that the programmer would like to capture a given error and handle the processing.

ErrorLevel: The ErrorLevel parameter is an integer type. This is the Message Level that VB Error Ease has determined for the given ErrorNumber. In addition Error Ease will use this number for logging purposes. The only modifications to this parameter that VB Error Ease will recognize is a setting of 0 which will notify VB Error Ease that the message is to be treated as a log only message. This has the same effect as using the LogOnly property with one addition. The addition of logging the actual information regarding the Message that invoked VB Error Ease via the ErrorNumber property. The only other allowable modification to this parameter is a value of 60-99, the Explicit result Messages. See the ErrorLevel property for more information.

ErrorMessage: The ErrorMessage parameter is a string type. This is the complete Message as will be presented and or logged byVB Error Ease. This parameter is provided in the event that the programmer would like to capture a given error and handle the processing himself. This parameter would provide the programmer with the verbage for the given message.

ErrorAction: The ErrorAction parameter is an integer type. This is an action flag provided to give the programmer a direct dialog with VB Error Ease. The value of 1 is set by default when entering the ErrorTrap event. This signifies that the active message should be presented to the application user, and it should be logged to the history log file. The only allowable overrides to this parameter are: Setting it to a 0 which signifies to Error Ease that the message should be ignored completely; Setting it to a 2 which signifies that the message should be logged to the history log file only and not displayed to the application user; Setting it to a 3 which signifies that the message should be displayed to the application user and not logged to the history log file. Any other settings besides the ones described above will yield the same result as a setting of a 1. CSERRLIT.TXT has been included with the Error Ease package which includes the mentioned literals as well as other literal values used for the VB Error Ease message handling process.

ErrorLogSet

The purpose of the ErrorLogSet event is to give the programmer a common coding procedure for setting basic information as it pertains to the current VB Error Ease handled message. This event is "fired" prior to any VB Error Ease action on or for the active Error Ease handled message. By utilizing this "stopping" point, the programmer can reduce the amount of code needed in each error block. A common usage for this event is to set common message related data. For example: By placing the following line of code in this event for Visual Basic application, the programmer can reduce the need to code this more than one time;
CSerr1.ErrorMessageVB = Error\$(CSerr1.ErrorNumber). This line of code will provide Error Ease with the necessary message verbage in the event that the message is not found in the Message Table.

The History Log File

A History Log File entry is created after a VB Error Ease handled message has been handled. The log entry is made immediately prior to the program control being returned to Visual Basic. If the entry being logged is the first one for a given log file, VB Error Ease will create the file at that point. The History Log File name and location are governed by the ErrorLogName property. The History Log File or files are created in either standard ASCII text format or comma delimited format for importing into a database. The output format is controlled by the LogFormat property. Logs created in ASCII text mode can be viewed using a simple editor such as the Windows Write application as well as the DOS Edit application. Logs created using the comma delimited feature can be import into any database package capable of importing comma delimited text. The comma delimited output is very useful for generating custom reports. See the LogFormat property for complete information regarding the available log output formats.

By setting the LogEnabled property to False, the History Log File can be disabled globally. By changing the ErrorAction value passed to via the ErrorTrap event to 3 the log entry for the given message can be bypassed.

In some instances an application may need to log messages to the History Log File, but not display the message to the user. An example of this may be if an application needs to log all transaction data for a given account number for future interrogation. There are several ways to do this; the first is to set the Message Level in the Message Table to a 0 to signify log only messages. A second way is to detect the given Message in the ErrorTrap event and change the ErrorLevel to a 0. If Error Ease does not need to interrogate the Message Table for this message, then the LogOnly property is the best choice.

Whats in the History Log File?

Every Error Ease handled message that is going to be logged is given a new log entry. By looking at a sample output log, which can be created by using the CSsample application, it is easy to recognize where the beginning and end of an entry.

Note: If large volumes of logging are to be done for a given application, keep in mind that the log file can become very large. It is a good practice to archive any needed log files to a backup medium and delete the logs from the application PC.

The following is an overview of the information within a History Log File entry:

Date/Time: This is the date and time that the Message was handled by VB Error Ease.

User ID: This is the ID passed via the ErrorUserID property.

Executable File Name:

This is the name of the application that has invoked the VB Error Ease OCX.

Module: This is the module name passed via the ErrorModule property.

Procedure: This is the procedure name passed via the ErrorProcedure property.

Message Number:

This is the ErrorNumber that invoked the VB Error Ease OCX. This is set to 0 for log only messages.

Message Level:

This is the ErrorLevel that has been determined for the given message. This is done using the Message Level from the Message Table or by using the ErrorLevelOVD property. In addition, this can be altered from within the ErrorTrap event. If the Message Level has been overridden, an additional line of text to that affect will follow in the log file.

Line Number:

This is the BASIC line number that was passed via the ErrorLine property.

User Message:

This is the actual message text that corresponds to the given message entry. If a message is an Explicit Result Message, text to that affect will be placed in this section of the log.

User Response:

This is the actual button selection the user has chosen upon being presented with a VB Error Ease invoked MessageBox. If the message is a log only message, text to that affect will be placed in this section of the log.

Application Provided Information:

This is the text that has been passed via the ErrorLogText property for the given message.

UserLogField1 through UserLogField5:

These five fields are free form log fields that are provided to the programmer for specialized reporting. UserLogTitle1 through UserLogTitle5 correspond to these generic log fields.

Frequently Asked Questions

Q. Do I have to have a message table in order to log messages to the history log file?

A. No. Logging is controlled by the ErrorLogName, LogEnabled, and LogFormat properties. This is independent of the Error Table features. The ErrorAction parameter within the ErrorTrap event can also be used to control logging behavior.

Q. What exactly does the 'OCX Code' check box option do during VBEASE code generation?

A. This option is only available if VBEASE detects the presence of a CSerr control within a given module. By choosing this option, VBEASE will add some standard logic to the ErrorLogSet event of the CSerr.OCX control. However, if VBEASE detects the existence of code in the ErrorLogSet event, VBEASE will not insert any further program code.

Q. If the same error, (i.e. 3022 - duplicates not allowed), needs to be handled in two different ways depending upon the procedure, what would be a suggested way to handle it?

A. This can be done a couple of ways; the first is to add a special line of code to the top of the error block of the specific procedure such as:

```
If Err = 3022 Then FRM_Main!CSerr1.ErrorLevelOVD = XX
```

where xx is a specific result message value. See ErrorLevel property for a complete definition of Explicit Result Messages. Then simply add the result value to the "CASE" block and Resume as required.

A better approach to this is to warehouse this type of special handling in the ErrorLogSet or ErrorTrap events in order to create and maintain a single point for specialized handling. This would be done similarly to the above example with the addition of a compare for the procedure name. This is found in the ErrorProcedure property of the CSerr.OCX. See the ErrorLogSet event within the CSSAMPLE program for a code sample.

How to Contact CASL s o f t



Sales and Support

n Telephone - (860) 585-8186

n Mailing Address - CASL s o f t

22 McKinley Avenue
Bristol, CT. 06010

n Email - rcasale@megahits.com

CASL s o f t

CASL s o f t - Sales/Support (860) 585-8186

Procedure List

List of all procedures for the current module being processed.

VB Error Ease

An enhanced error handling tool.

