REFERENCE POSTER

NI TestStand™ User Interface Controls

Overview

The NI TestStand User Interface (UI) Controls are a set of ActiveX controls that implement the common functionality operator interfaces and custom sequence editors need in order to display, execute, edit, and debug test sequences. These ActiveX controls greatly reduce the amount of source code a user interface application requires.

The TestStand UI Controls are divided into two categories: manager controls and visible controls. This poster describes the visible controls—text display, selector, command, and view—as well as the manager controls.

National Instruments recommends that you use these controls to develop your user interface applications. Refer to Chapter 9, *Creating Custom User Interfaces*, in the *NI TestStand Reference Manual* for more information about developing applications with the TestStand UI Controls.

The TestStand UI Controls greatly reduce the need for an application to directly call the TestStand API. However, you can still call the TestStand API directly on objects you create or obtain from the TestStand UI Controls properties, methods, or events. If you call the TestStand API from a user interface that uses the TestStand UI Controls, follow the guidelines listed in the *Using the TestStand API with TestStand UI Controls* section of Chapter 9, *Creating Custom User Interfaces*, in the *NI TestStand Reference Manual*.

National Instruments, NI, ni.com, NI TestStand, and LabVIEW are trademarks of National Instruments Corporation. Refer to the *Terms of Use* section on ni.com/legal for more information about National Instruments trademarks. Other product and company names mentioned herein are trademarks or trade names of their respective companies. For patents covering National Instruments products, refer to the appropriate location: **Help»Patents** in your software, the patents.txt file on your CD, or ni.com/patents.

© 2003–2007 National Instruments Corporation. All rights reserved. Printed in Ireland.

373423B-01 Apr07

Legend

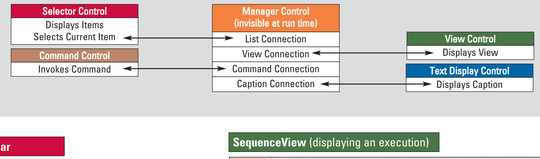
Control ♦	A visible or invisible ActiveX component that you use on user interface windows.
Object	A programmable software component representing an instance of a class. You interact with the object using the methods, properties, and events that the class defines.
Property	A container of information, which stores and maintains a setting or attribute of an object. A property can contain a single value, an array of values of the same type, or no value at all.
(Read Only)	Properties that can be read but not changed.
Method	Performs an operation or function on an object.
Event	Method that an object invokes to inform a client application of a status or state change.
Collection	An object that contains zero or more objects of the same class.
Collection Class	Denotes a class that has a corresponding collection class. The top (lighter) text represents the collection. The lower (darker) text represents the type of object the collection contains. Collection class properties and methods are not listed on this poster.
◄►	Denotes a connection between a manager control and another control. Information is shared between the two controls. For example, a Button control instructs a manager control to execute a command, whereas the manager control determines when the Button control is dimmed.

Note: Some properties and methods from previous versions of TestStand are obsolete and are not listed on this poster.

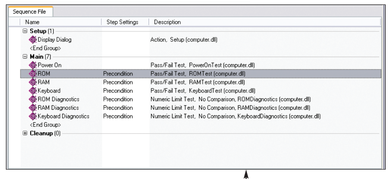
Connection Types

Connect a manager control to a visible control to automatically display sequences or reports, present a list of items to the user, invoke an application command, or display information about the current state of the application. When you connect controls, your application does not need the majority of the source code you would usually write in order to update the user interface and respond to user input.

Connections from manager controls to visible controls are many-to-one. Therefore, you can make the same connection from a manager control to multiple visible controls. The specific connections you can make depend on the type of manager control and visible control that you are connecting, as shown on the right. Specific examples of these connections are illustrated below.

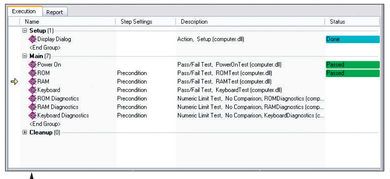


SequenceView (displaying a sequence file)

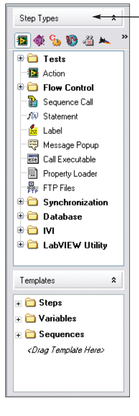


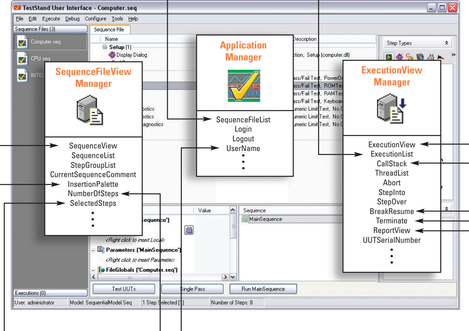
ListBar

SequenceView (displaying an execution)



InsertionPalette

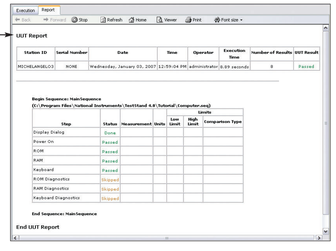




ListBox

Button

ReportView



StatusBar

Manager Controls

Manager controls call the TestStand API to perform tasks such as loading files, launching executions, and editing and retrieving sequence information. Manager controls also notify you when application events occur, such as when a user logs in, an execution reaches a breakpoint, or a user changes the file or sequence that they are viewing. These controls are visible at design time but invisible at run time. Use an Application Manager control to perform basic operations that are necessary to use the TestStand Engine in your application. Your application must have a single Application Manager control that exists for the duration of the application. Use a SequenceFileView Manager control to manage how visible TestStand UI Controls view, edit, and interact with a selected sequence file. Your application must have one SequenceFileView Manager control for each location, such as a window, form, or panel, in which you display a sequence file or allow a user to select the current sequence file. Use an ExecutionView Manager control to manage how visible TestStand UI Controls view and interact with a selected TestStand execution. Your application must have one ExecutionView Manager control for each location, such as a window, form, or panel, in which you display an execution or allow a user to select the current execution.

♦ApplicationMgr

Properties
ApplicationWillExitOnStart (Read Only)
AutomaticallyReloadModifiedFiles
BreakOnFirstStep
BreakOnSequenceFailure
BreakOnStepFailure
CommandLineArguments (Read Only)
CommandLineChangeEditMode
ConfigFile (Read Only)
ConfigFilePath
ConfigurationEntryPoints (Read Only)
Connections (Read Only)
CreateEmptySequenceFileOnStart
CurrentUIMessage (Read Only)
EditModeShortcutKey
EditModeShortcutModifier
EditReadOnlyFiles
Executing (Read Only)
ExecutionEntryPoints (Read Only)
Executions (Read Only)
IsEditor
IsShuttingDown (Read Only)
IsStarted (Read Only)
LoginLogoutRunning (Read Only)
LoginOnStart
LoginRunning (Read Only)
LogoutClosesSeqFilesAndExecs
LogoutRunning (Read Only)
MakeStepNamesUnique
MRUFiles (Read Only)
ProcessCommandLine
PromptForOverwrite
ReloadModifiedFilesInterval
ReloadSequenceFilesOnStart
SaveOnClose
SequenceFiles (Read Only)
UndoStack
UserData
UseStepListConfigurations

Methods
AddCommandLineArgumentsHelp
BeginEdit
CanEdit
CloseAllExecutions
CloseAllSequenceFiles
CloseExecution
CloseSequenceFile
ConnectAdapterList
ConnectCaption
ConnectCommand
EndEdit
GetAutoCloseExecution
GetCaptionText
GetCommand
GetEngine
GetExecutionViewMgr
GetImageName
GetModelFile
GetRunState
GetSequenceFileViewMgr
GetTerminationState
GetVisible
LocalizeAllControls
Login
continued, next column

Logout
NewCommands
OpenSequenceFile
OpenSequenceFileDialog
OpenSequenceFilesDialog
RaiseError
Refresh
RefreshAllViewMgrs
RefreshFile
ReloadConfigFile
ReloadFile
ReloadModifiedSequenceFiles
SetAutoCloseExecution
SetVisible
ShutDown
Start

Events
AfterUIMessageEvent
BeginEdit
Break
BreakOnRunTimeError
CanEdit
DisplayExecution
DisplayReport
DisplaySequenceFile
DropFile
EditModeChanged
EndEdit
EndExecution
ExecutionClosed
ExitApplication
PostCommandExecute
PreCommandExecute
ProcessUserCommandLineArguments
QueryCloseExecution
QueryCloseSequenceFile
QueryReloadSequenceFile
QueryShutdown
RefreshWindows
ReportError
SequenceFileClosed
SequenceFileClosing
SequenceFileOpened
ShutDownCancelled
ShutDownCompleted
StartExecution
UIMessageEvent
UserChanged
UserMessage
Wait

Executions

Note: The Executions object is a collection that contains Execution objects. The Execution object is described on the *NI TestStand API Reference Poster.*

Properties
Count (Read Only)
Item (Read Only)
NumIncomplete (Read Only)
NumPassed (Read Only)
NumRunning (Read Only)

Method
ResumeAll

♦ExecutionViewMgr

Properties
ApplicationMgr (Read Only)
ConfigurationEntryPoints (Read Only)
Connections (Read Only)
Execution
ExecutionEntryPoints (Read Only)
ReplaceExecutionOnClose
RunState (Read Only)
SelectedPropertyObjects (Read Only)
SelectedSteps (Read Only)
SequenceContext
StepGroupMode
TerminationState (Read Only)
Thread
UserData

Methods
AbortExecution
BreakExecution
BuildEditArgs
BuildInteractiveArgs
ConnectCallStack
ConnectCaption
ConnectCommand
ConnectExecutionList
ConnectExecutionView
ConnectImage
ConnectNumeric
ConnectReportView
ConnectThreadList
ConnectVariables
GetCaptionText
GetCommand
GetImageName
GetNumericValue
LoopOnSelectedSteps
NewEditContext
Refresh
RefreshStep
RefreshStepEx
RestartExecution
ResumeExecution
RunSelectedSteps
TerminateExecution

Events
Break
ContextChanged
DisplayReport
EndExecution
ExecutionChanged
PropertyObjectSelectionChanged
RefreshWindow
RunStateChanged
SelectionChanged
TerminationStateChanged
ThreadChanged
Trace
UserMessage

♦SequenceFileViewMgr

Properties
ApplicationMgr (Read Only)
ConfigurationEntryPoints (Read Only)
Connections (Read Only)
ExecutionEntryPoints (Read Only)
ReplaceSequenceFileOnClose
SelectedPropertyObjects (Read Only)
SelectedSequences (Read Only)
SelectedSteps (Read Only)
Sequence
SequenceFile
StepGroup
StepGroupMode
UndoStack (Read Only)
UserData

Methods
BuildEditArgs
BuildInteractiveArgs
ConnectCaption
ConnectCommand
ConnectImage
ConnectInsertionPalette
ConnectSequenceFileList
ConnectSequenceList
ConnectSequenceView
ConnectStepGroupList
ConnectVariables
GetCaptionText
GetCommand
GetImageName
LoopOnSelectedSteps
NewEditContext
Refresh
RefreshStep
RefreshStepEx
Run
RunSelectedSteps
SetSequenceAndGroup

Events
PropertyObjectSelectionChanged
RefreshWindow
SelectionChanged
SequenceChanged
SequenceFileChanged
SequenceSelectionChanged
StepGroupChanged

SequenceFiles

Note: The SequenceFiles object is a collection that contains SequenceFile objects. The SequenceFile object is described on the *NI TestStand API Reference Poster.*

Properties
Count (Read Only)
Item (Read Only)

Methods
Note: This object has no methods.

Selector Controls

Connect a SequenceFileView Manager or ExecutionView Manager control to a ComboBox or ListBox control to view or select a list of files, sequences, step groups, executions, threads, or stack frames. Use a ListBar control to display multiple pages where each page contains a list of items. Connect a SequenceFileView Manager or ExecutionView Manager control to a ListBar page to view and select a list of items. Use the ComboBox control to connect to and display the ExecutionList, CallStack, and ThreadList in the ExecutionView Manager control and to connect to and display the SequenceFileList, SequenceList, and StepGroupList in the SequenceFileView Manager control.

♦ CheckBox

Properties
BackColor
Caption
Checked
Enabled
Font
FontSource
ForeColor
hWnd (Read Only)
Image
ImageAlign
MaskColor
ScaleWithDPI
Style
TextAlign
UseMaskColor

Method
Localize

Events
Click
ConnectionActivity
KeyDown
KeyPress
KeyUp
MouseDown
MouseMove
MouseUp

♦ ComboBox

Properties
Count (Read Only)
Enabled
Font
FontSource
HeightWithDropDown
hWnd (Read Only)
IconSize
ItemIndex
MouseIcon
MousePointer
ReadOnly
ScaleWithDPI
ShowImages
Style
ToolTipVisible

Method
GetItemText

Events
Change
Click
ConnectionActivity
DropDown
KeyDown
KeyPress
KeyUp

♦ ListBar

Properties
BackColor
Borders (Read Only)
ButtonFont
ButtonFontSource
ButtonStyle
ButtonTextColor
CurrentPage
hWnd (Read Only)
IconSize
PageFont
PageFontSource
Pages (Read Only)
PageStyle
PageTextColor
ScaleWithDPI
ShowItemCount
ShowItemTipStrips
ShowScrollBar

Methods
HitTest
Localize

Events
BorderDragged
ConnectionActivity
CreateContextMenu
CurPageChanged
CursorMoved
DblClick
KeyDown
MouseDown
MouseMove
MouseUp

ListBarPages ListBarPage

Properties
Caption
Cursor
Enabled
Index (Read Only)
Items (Read Only)
Name

Methods
Note: This object has no methods.

ListBarPageItems ListBarPageItem

Properties
Caption
IconName
Index (Read Only)
ToolTipText

Methods
Note: This object has no methods.

♦ ListBox

Properties
AutosizeColumns
BackColor
Borders (Read Only)
CanEditLabel (Read Only)
Columns
ColumnSet (Read Only)
Count (Read Only)
EditingFlags
Enabled
Font
FontSource
ForeColor
hWnd (Read Only)
ItemIndex
MouseIcon
MousePointer
ScaleWithDPI
ShowHeaders
ToolTipVisible

Methods
EditLabel
GetItemText
HitTest

Events
BorderDragged
Click
ColumnsResized
ConnectionActivity
CreateContextMenu
DblClick
KeyDown
KeyPress
KeyUp
MouseDown
MouseMove
MouseUp

ListBoxColumns ListBoxColumn

Properties
AutoSizing
Caption (Read Only)
Index (Read Only)
Width

Methods
Note: This object has no methods.

View Controls

Connect a SequenceFileView Manager or an ExecutionView Manager control to a SequenceView control to display the steps of a sequence from a selected file or execution. Connect an ExecutionView Manager control to a ReportView control to display the report for the selected execution.

Connect a SequenceFileView Manager control to an InsertionPalette control for inserting steps in sequences.

Connect a SequenceFileView Manager or an ExecutionView Manager control to a VariablesView control to display variables and properties.

♦InsertionPalette

Properties
AllowEditing
Borders (Read Only)
Enabled
hWnd (Read Only)
Pages (Read Only)
PaletteLayout
ScaleWithDPI

Methods
Note: This object has no methods.

Events
BorderDragged
ConnectionActivity
KeyDown

InsertionPalettePages InsertionPalettePage

Properties
Expanded
Index (Read Only)
SplitterRatio
Visible

Methods
Note: This object has no methods.

♦ReportView

Properties
Borders (Read Only)
ButtonsVisible
LargeIcons
ScaleWithDPI
TextReportBackColor
TextReportColor
TextReportFont
TextReportFontSource
ToolBarTextStyle
ToolBarVisible

Methods
GetHTMLCtrl
GetRichEditCtrl
PrintReport
UpdateFromExecution

Events
BorderDragged
ConnectionActivity

♦SequenceView

Properties
AutoSizeColumns
BackColor
BlockDisplayOptions
Borders (Read Only)
CanEditLabel (Read Only)
Columns (Read Only)
CommentsColor
CommentsFont
CommentsFontSource
CommentsOffset
Cursor
EditingFlags
FocusIndex
HeaderFont
HeaderFontSource
HorizLines
hWnd (Read Only)
IconSize
ItemBackColorExpression
ItemTextColorExpression
LinesColor
MaxCommentsHeight
RoundItemRects
ScaleWithDPI
ShadeAlternateColumns
ShowComments
ShowCommentsBars
ShowItemTipStrips
StepIconsEnabled
TextColor
TextFont
TextFontSource
VertLines

Methods
BeginUpdate
EditLabel
EndUpdate
HitTest
ItemIndexToStep
Localize
StepToItemIndex

Events
BorderDragged
ColumnResized
ConnectionActivity
CreateContextMenu
CursorMoved
DblClick
KeyDown
MouseDown
MouseMove
MouseUp

SeqViewColumns SeqViewColumn

Properties
AutoSizing
BackColorExpression
Caption
Expression
Index (Read Only)
Name
TextColorExpression
Type
Visible
Width

Methods
Note: This object has no methods.

♦VariablesView

Properties
Borders (Read Only)
Enabled
hWnd (Read Only)
ScaleWithDPI

Methods
Note: This object has no methods.

Events
BorderDragged
ConnectionActivity
KeyDown

Text Display Controls

An ExpressionEdit control allows you to edit a TestStand expression with the convenience of syntax coloring, popup help, and statement completion.

Connect a manager control to an ExpressionEdit control in a custom sequence editor to allow users to view and edit expressions. In a user interface, connect a manager control to a read-only ExpressionEdit control to automatically display text information about the application state, such as the pathname of the selected sequence file or the name of the current user.

Connect a manager control to panes of a StatusBar control to display text, image, or progress information about the application state. Connect a manager control to a Label control to automatically display text information about the application state in the label.

♦ExpressionEdit

Properties
AllowEmpty
AutoCompletionHwnd (Read Only)
AutomaticallyPrefixVariables
BackColor
Borders (Read Only)
BrowseExprDialogOptions
BrowseExprDialogTitle
Buttons (Read Only)
ComboBoxItems (Read Only)
Context
DisplayFormattedValue
DisplayText
DropDownListHwnd (Read Only)
Enabled
Engine
ErrorCheck
Font
FontSource
FunctionTipHwnd (Read Only)
HideSelection
hWnd (Read Only)
MaxLength
MouseIcon
MousePointer
Multiline
NumericFormat
ReadOnly
RequiredNamedTypeArrays (Read Only)
RequiredNamedTypes (Read Only)
RequiredTypes
ScaleWithDPI
ScrollBars
SelLength
SelStart
SelText
ShowDisplayNameWhenInactive
Style
SyntaxHighlightingEnabled
Text
TextLength (Read Only)
TextType
WantReturn
WordWrap

Methods
CheckExpression
DisplayBrowseExprDialog
DisplayError
Evaluate
GetAdditionalEvaluationConstants
Localize
SelectAll
SetAdditionalEvaluationConstants

Events
BorderDragged
BrowseExprDialogClosed
BrowseExprDialogOpened
continued, next column

ButtonClick
Change
CheckExpression
Click
ConnectionActivity
ContextChanged
ContextMenuItemClick
CreateContextMenu
DblClick
DropDown
InsertComboBoxItem
KeyDown
KeyPress
KeyUp
MouseDown
MouseMove
MouseUp
SelChange

ExpressionEditButtons ExpressionEditButton

Properties
BackColor
ContextMenuItemCaption
Enabled
hWnd (Read Only)
Icon
Kind
ShortcutKey
ShortcutModifier
Style
ToolTipText
Visible

Method
DnClick

ExpressionEditComboBoxItems ExpressionEditComboBoxItem

Properties
DisplayName
Icon
Value

Methods
Note: This object has no methods.

♦ Label

Properties
Alignment
AutoSize
BackColor
BorderStyle
Caption
Enabled
Font
FontSource
ForeColor
MouseIcon
continued, next column

MousePointer
ScaleWithDPI
UseMnemonic
WordWrap

Method
Localize

Events
Change
Click
ConnectionActivity
MouseDown
MouseMove
MouseUp

♦ StatusBar

Properties
AutoSize
Font
FontSource
hWnd (Read Only)
IconSize
Panes (Read Only)
ScaleWithDPI
ShowResizingGrip
ShowTopDivider

Methods
HitTest
Localize
ShowPanes

Events
ConnectionActivity
DblClick
MouseDown
MouseMove
MouseUp
Resized

StatusBarPanes StatusBarPane

Properties
Caption
IconName
Index (Read Only)
Name
ProcessPercent
Style
TextAlignment
ToolTipText
UseAvailableSpace
Visible
Width

Methods
Note: This object has no methods.

Command Controls

Connect a manager control to a Button control to specify that the button performs a common user interface command. A Command object represents a standard action that you can initiate in a TestStand application, such as opening a sequence file or executing a set of selected steps in a sequence. Command objects make it simpler for an application to implement a user interface element that performs a standard action. An EntryPoint object represents an Execution or Configuration entry point. You can execute the entry point sequence using the Run, RunSelectedSteps, or LoopOnSelectedSteps methods. Use the ConfigurationEntryPoints and ExecutionEntryPoints methods of the SequenceFileView Manager and ExecutionView Manager controls to retrieve an EntryPoints object that is a collection of entry points. Refer to the *Menus and Menu Items* section of Chapter 9, *Creating Custom User Interfaces*, of the *NI TestStand Reference Manual* for information about how to create menu items that perform common user interface commands.

♦ Button

Properties
ActionStyle
AutoSizing
BackColor
Caption
Enabled
Font
FontSource
ForeColor
hWnd (Read Only)
Image
ImageAlignment
MaskColor
MouseIcon
MousePointer
ScaleWithDPI
ShowImage
ShowText
Style
TextAlignment
TextImageRelation
ToolTipText
UseMaskColor
UseMnemonic
Value

Methods
DoClick
Localize

Events
Click
ConnectionActivity
KeyDown
KeyPress
KeyUp
MouseDown
MouseMove
MouseUp

Commands Command

Properties
Enabled
EntryPoint (Read Only)
EntryPointIndex (Read Only)
ExecutionViewMgr
IconName
IsSeparator
IsToggle
Kind (Read Only)
LVShortcutKey
continued, next column

LVShortcutModifier
SequenceFileViewMgr
ShortcutKey
ShortcutModifier
SubsidiaryCommands
ToggleState
UserData
UserObject
Visible

Methods
Execute
GetDisplayName
SetDisplayName

EntryPoints EntryPoint

Properties
Enabled (Read Only)
InteractiveExecutionAllowed (Read Only)
Name (Read Only)
Sequence (Read Only)

Methods
LoopOnSelectedSteps
Run
RunSelectedSteps

Miscellaneous

A Borders object specifies the frame and draggable borders that surround a control. If your application handles the BorderDragged event for the control, users can drag the borders to resize or move the control. Refer to the *NI TestStand Help* for information about additional Support UI classes, properties, methods, and events.

Borders

Properties
BorderDraggedEventEnabled
BorderEdgeStyle
BorderWidth
FrameEdgeStyle
FrameInsideBorders
FrameVisible
VisibleBorders

Method
GetFrameThickness