VIP for Dictionary-IV®
Reference Manual

THOROUGHBRED
software international, inc

Version 8.7.1

285 Davidson Ave., Suite 302 • Somerset, NJ 08873-4153
Telephone: 732-560-1377 • Outside NJ 800-524-0430
Fax: 732-560-1594

Internet address: **http://www.tbred.com**

Published by:
Thoroughbred Software International, Inc.
285 Davidson Ave., Suite 302
Somerset, New Jersey 08873-4153

Document Number: VIP8.7.1M01

VIP FAMILY OF PRODUCTS

Thoroughbred VIP, (Visual Image Presentation), is a family of software products that enables Microsoft Windows workstations to access host-based applications. Depending upon the software package you purchase you will receive one or all of the following.

VIP for Dictionary-IV

IMPORTANT: The VIP GUI Screen Editor and Toolbar Definition Maintenance are not supported on the Windows Vista, Windows Server 2008, and Windows 7.

Connects from a Windows workstation to a host application. Dictionary-IV Screens, Menus, Messages, Help, Views, Reports and Query definitions are automatically converted to graphical objects providing the user with a GUI presentation of a character based host application.

When running VIP for Dictionary-IV you will have all the functionality described in the Dictionary-IV Developer and User guides. In addition you will be able to:

- Run external applications from a graphical screen, toolbar, or host application code.
- Control view fonts and color.
- Control special screen field colors.
- Create custom graphical screens.
- Create templates for graphical screens.
- Use your right mouse button in definition view fields to access lookups, connects, help and data element properties.
- Support for reloading local cache on workstations.

WorkStation Manager

WorkStation Manager is an integral part of the VIP family of products. It provides customization in the use of and the look of TbredComm and VIP for Dictionary-IV and provides the ability to add Dictionary-IV based menus to the Windows Start Menu.

WorksStation Manager is included with all the VIP products.

Gateway for Windows

Thoroughbred Gateway for Windows provides a DDE communication path between the host applications and Windows-based software using existing DDE interfaces.

Gateway for Windows is included with all the VIP products.

TbredComm

TbredComm provides a 32-bit TCP/IP and a Serial interface for optimum network communication and terminal emulation. TbredComm is required when running VIP on a remote host.

- Support the specific needs of the Thoroughbred Environment.
- Set default and custom AutoLogons.
- Use various TCP/IP protocols including raw, telnet and SSH.
- Set null and secured passwords.
- Use mouse aware capabilities for your 3GL applications
- Use mouse aware capabilities for your character-based Dictionary-IV and OPENworkshop applications. Starting with release 8.5.1, both Dictionary-IV and OPENworkshop are mouse aware.
- Copy and Paste across applications.
- Have the ability to allow any file on the client that can be executed from the Windows Run window, to be opened and executed from within TbredComm.
- Use slave printing and the Windows print manager.
- Support 24 function keys
- Set scrolling
- Use Print Screen
- Use Font Selections
- Use Dictionary-IV color settings
- Have the ability to open and execute files located on a windows workstation from your application code. If a file can be opened or launched from the Windows Run dialog, it can be opened by your application code using TbredComm.
- Provides 132-column support using the Basic wide-screen mnemonic. Requires Basic 8.5.0 for the correct terminal table. For more information on Basic mnemonics, please refer the Thoroughbred Basic Developer Guide.

TbredComm is sold separately. It is available for both site license installations and per user (activation) installations. TS ReportServer is sold separately. Please contact your sales representative for more information.

TS ReportServer

TS ReportServer allows Thoroughbred-based reports to be displayed graphically, printed, and saved to a Windows based workstation directly from a Thoroughbred application. It enables the user to change the font, define the printer, search and find keywords, and copy selected portions or the entire report to a saved file or other Windows-based application. The following is a summary of TS ReportServer features

- Scroll forward and backward through report data.
- Highlight header information.
- Print to any printer accessible through the Windows Print Manager.
- Print to any Fax device.
- Zoom in and out for flexible viewing.
- Change font sizes and styles.
- Save as .txt file for export to other Windows applications.
- On-line storage of reports for future viewing and printing.
- Find any selected string data.
- Print locally or at a remote site.
- Define split screens for viewing multiple sections of a report.

TS ReportServer is sold separately. Please contact your sales representative for more information.

TS ChartServer

TS ChartServer allows you to create simple multi-dimensional charts quickly from any Thoroughbred Development Environment. TS ChartServer supports pie charts, bar charts, stacked bar charts, graphical plots, and area charts.

TS ChartServer is sold separately. Please contact your sales representative for more information.

IMPORTANT TS ChartServer is not supported on Windows Vista, Windows Server 2008, and Windows 7.

PATH DEFINITIONS AND CONVENTIONS

The paths noted in this manual vary depending upon the operating system.

vip-data-path is used to indicate the path where files requiring read and write access are located.

- On Windows Vista/Windows Server 2008/Windows 7 the required path is C:\ProgramData\Thoroughbred Software\TSI.

- On Windows Server 2000/Windows XP/Windows Server 2003 the default path is C:\Program Files\TSI.

vip-user-path is used to indicate the path where user specific files requiring read and write access are located.

- On Windows Vista/Windows Server 2008/Windows 7 the required path is C:\Users\tbred2\AppData\Roaming\Thoroughbred Software.

- On Windows Server 2000/Windows XP/Windows Server 2003 this is the same as the *vip-data-path.*

vip-program-path is used to indicate the path where executable and some dll files are located.

- On all Window platforms the default path is C:\Program Files\TSI\Bin.

vip-system-path is used to indicate the path where Thoroughbred ini files and other system files that require write access are located.

- On Windows Vista/Windows Server 2008/Windows 7 the required path is C:\ProgramData\Thoroughbred Software.

- On other Window platforms this path may vary, typically it is C:\Documents and Settings\All Users\Application Data\Thoroughbred Software.

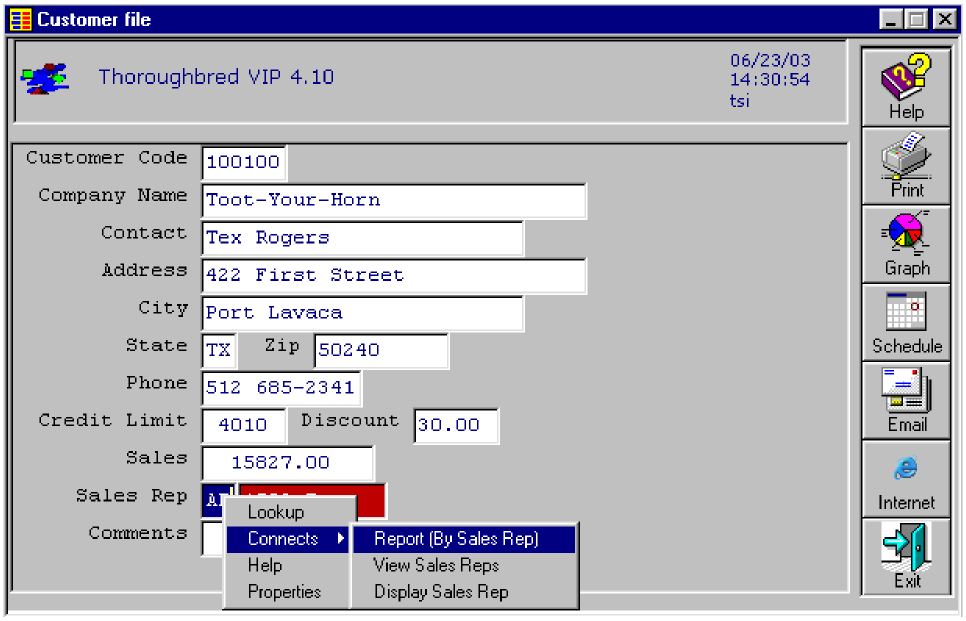
GRAPHIC DEFINITIONS IN DICTIONARY-IV

The following section provides a brief description of the graphic version of Dictionary-IV definitions.

Screens

Graphic screens have all the functionality described in the Dictionary-IV Developer Guide and User guides.

Below is an example of a custom graphic screen.



Customer file

Thoroughbred VIP 4.10

06/23/03
14:30:54
tsi

Customer Code 100100

Company Name Toot-Your-Horn

Contact Tex Rogers

Address 422 First Street

City Port Lavaca

State TX Zip 50240

Phone 512 685-2341

Credit Limit 4010 Discount 30.00

Sales 15827.00

Sales Rep AI

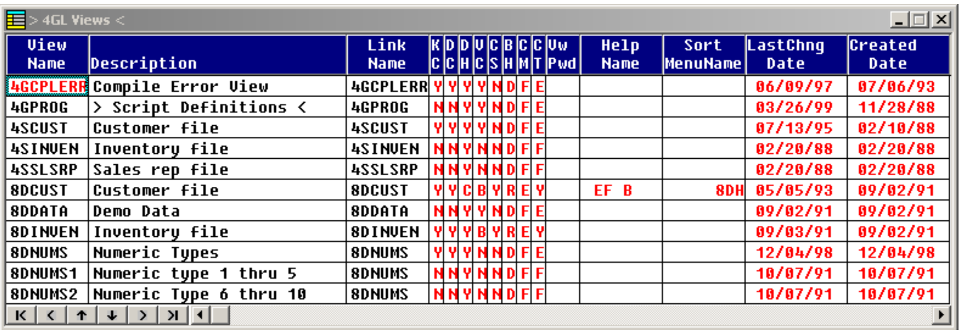
Comments

For more information on creating and using graphical screens please see the Create Graphic Presentations section and the Graphic Screen Designer section.

Views

Graphic views have all the functionality described in the Dictionary-IV Developer Guide and User guides.

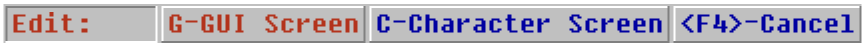
The following is an example of a graphic view:



View		Link	K	D	D	V	C	B	C	C	Vw	Help	Sort	LastChng	Created
Name	Description	Name	C	C	H	C	S	H	M	T	Pwd	Name	MenuName	Date	Date
4GCPLERR	Compile Error View	4GCPLERR	Y	Y	Y	Y	N	D	F	E				06/09/97	07/06/93
4GPROG	> Script Definitions <	4GPROG	N	N	Y	Y	N	D	F	E				03/26/99	11/28/88
4SCUST	Customer file	4SCUST	Y	Y	Y	Y	N	D	F	E				07/13/95	02/10/88
4SINVEN	Inventory file	4SINVEN	N	N	Y	N	N	D	F	F				02/20/88	02/20/88
4SSLSRP	Sales rep file	4SSLSRP	N	N	Y	N	N	D	F	F				02/20/88	02/20/88
8DCUST	Customer file	8DCUST	Y	Y	C	B	Y	R	E	Y		EF B	8DH	05/05/93	09/02/91
8DDATA	Demo Data	8DDATA	N	N	Y	Y	N	D	F	E				09/02/91	09/02/91
8DINVEN	Inventory file	8DINVEN	Y	Y	Y	B	Y	R	E	Y				09/03/91	09/02/91
8DNUMS	Numeric Types	8DNUMS	Y	Y	Y	N	N	D	F	E				12/04/98	12/04/98
8DNUMS1	Numeric type 1 thru 5	8DNUMS	N	N	Y	N	N	D	F	F				10/07/91	10/07/91
8DNUMS2	Numeric Type 6 thru 10	8DNUMS	N	N	Y	N	N	D	F	F				10/07/91	10/07/91

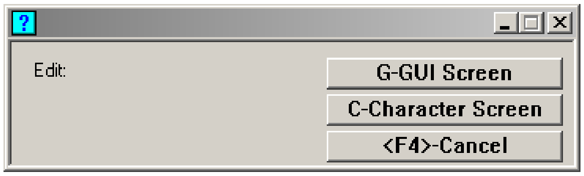
Messages

Graphic messages have all the functionality described in the Dictionary-IV Developer and User guides. VIP supports all Dictionary-IV and OPENworkshop message types. Messages can be displayed in the Status Bar:



Edit: G-GUI Screen C-Character Screen <F4>-Cancel

or in a Dialog Box.

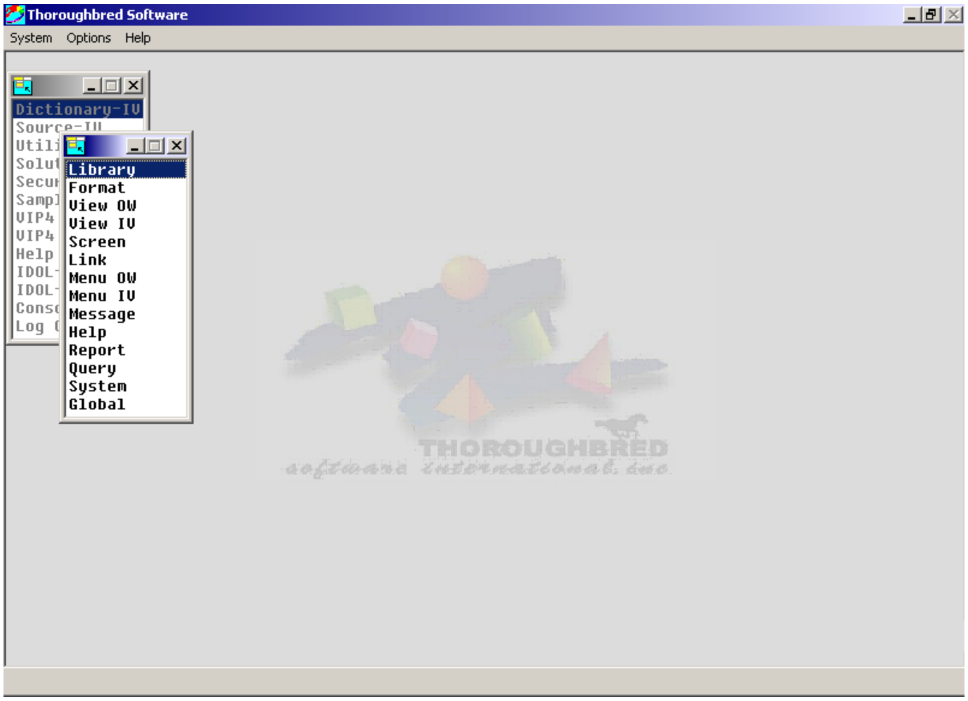


Edit:

Where appropriate, VIP automatically adds an input control and buttons to the message control.

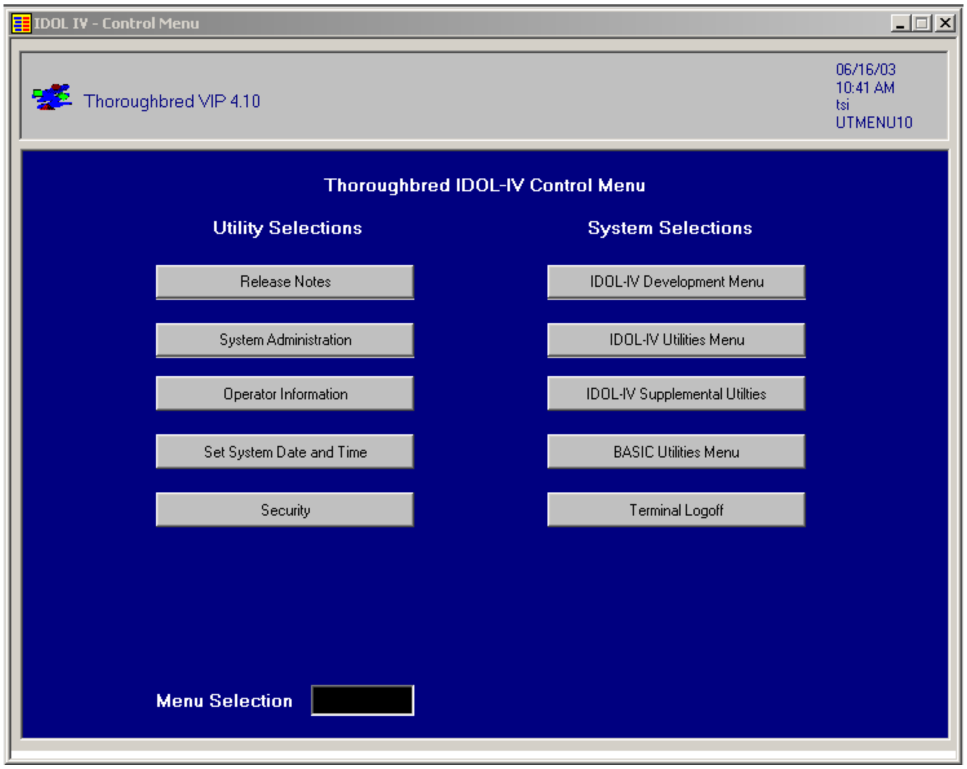
Menus

Graphic OPENworkshop (pop-up) menus have all the functionality described in the Dictionary-IV Developer and User guides. They are presented as a list within a GUI window.



For more information please see the Create Graphic Presentations section.

Graphic Dictionary-IV menus have all the functionality described in the Dictionary-IV Developer and User guides. Dictionary-IV menus are presented using graphical screen definitions. Graphical Dictionary-IV menus support all the features and functionality of a graphical screen definition.


IDOL IV - Control Menu

Thoroughbred VIP 4.10

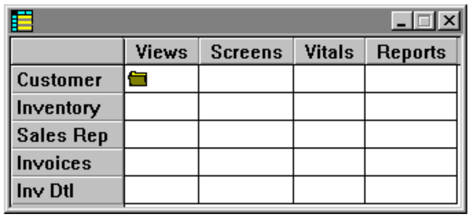
06/16/03
10:41 AM
tsi
UTMENU10

Thoroughbred IDOL-IV Control Menu

For more information please see the Create Graphic Presentation section and the Graphic Screen Designer Section.

Matrix Menus

Graphic matrix menus have all the functionality described in the OPENworkshop manual. Below is an example of a graphic matrix menu.



	Views	Screens	Vitals	Reports
Customer				
Inventory				
Sales Rep				
Invoices				
Inv Dtl				

For more information please see the Create Graphic Screen Presentation section.

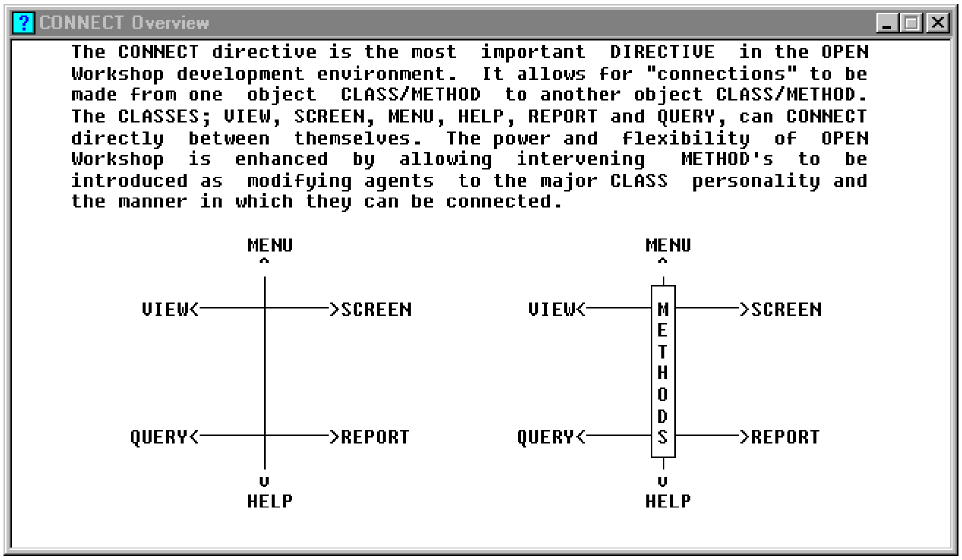
On-line Help

Graphic Help has all the functionality described in the Dictionary-IV Developer and User guides. The various help types will be displayed in appropriate graphical controls.

- Simple on-line help displays in a Windows text control.
- Help Index (.HI.) displays in a graphical list, see OPENworkshop menus.

- Pop-up Menus (.HM., .MN.) displays in a graphical list, see OPENworkshop menus.
- Function Key (.HF.) displays in a display in a Windows text control.

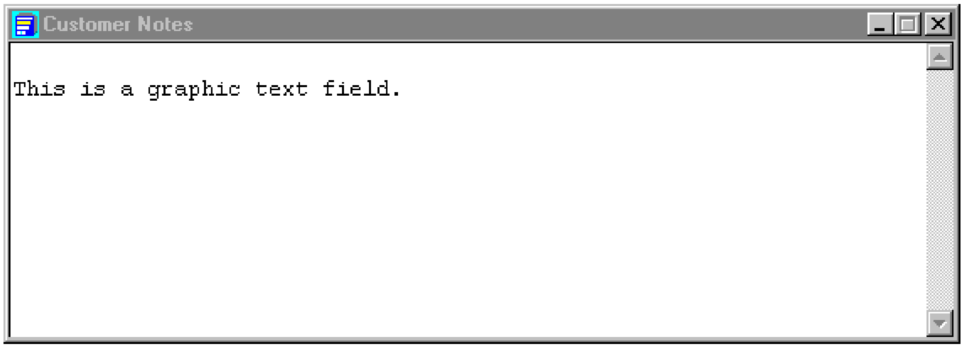
Below is an example of graphic simple on-line help:



The CONNECT directive is the most important DIRECTIVE in the OPEN Workshop development environment. It allows for "connections" to be made from one object CLASS/METHOD to another object CLASS/METHOD. The CLASSES; VIEW, SCREEN, MENU, HELP, REPORT and QUERY, can CONNECT directly between themselves. The power and flexibility of OPEN Workshop is enhanced by allowing intervening METHOD's to be introduced as modifying agents to the major CLASS personality and the manner in which they can be connected.

Text Field

Graphic Text Fields display in the Windows text control. Below is an example of a graphic text field.



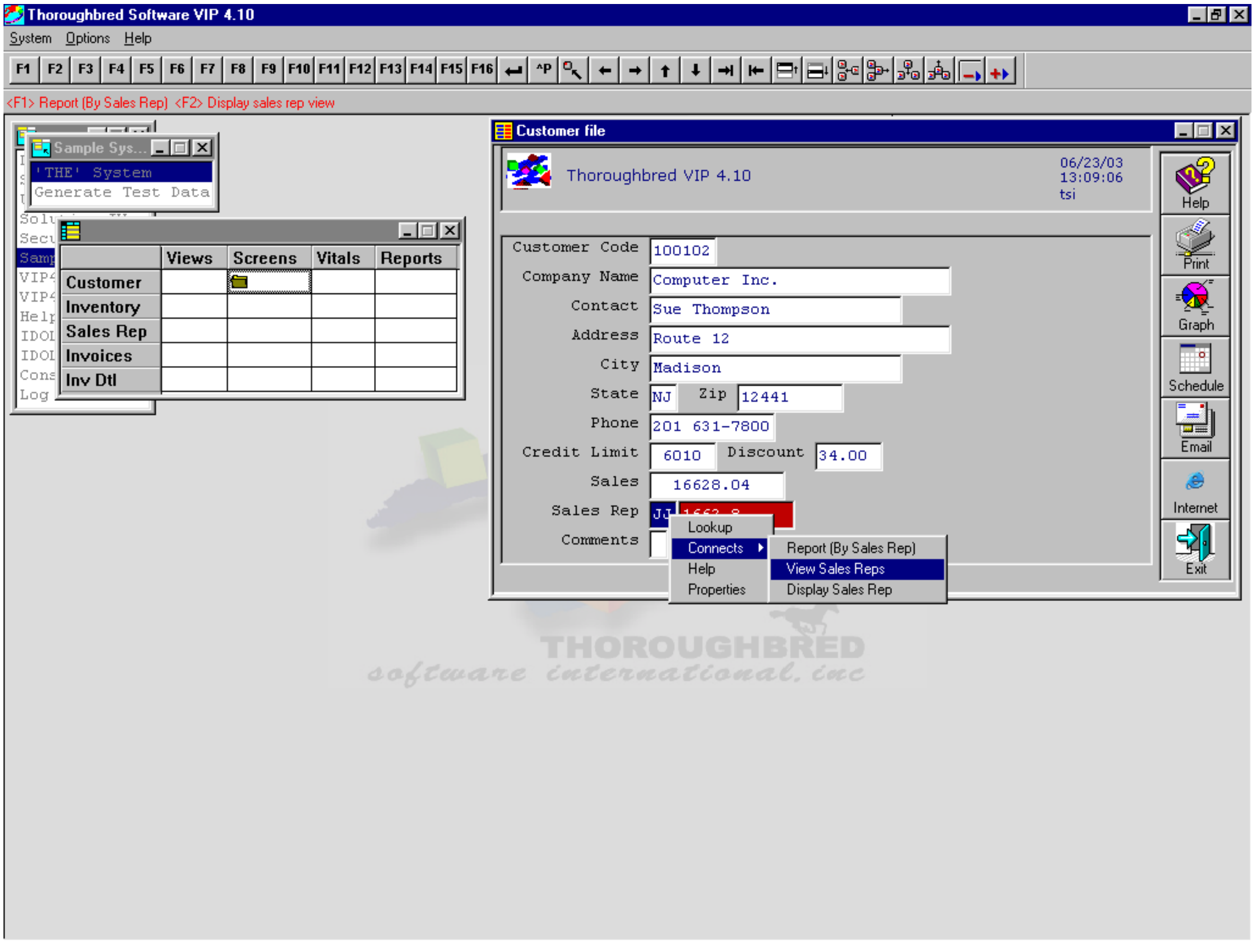
This is a graphic text field.

CREATE GRAPHIC PRESENTATIONS

With VIP enabled, the host application code executes exactly as it does in character mode. Business rules and logic, data integrity, data element and file definitions remain under the control of the host application code. When the host application code references a Dictionary-IV definition, VIP will automatically convert it to a graphical presentation on the client. The host application code is independent from the presentation style.

This section describes how to use and customize the Dictionary-IV graphical objects that are applied to a character application.

All VIP graphical objects are displayed within the main VIP Server Window.



<F1> Report (By Sales Rep) <F2> Display sales rep view

'THE' System
Generate Test Data

	Views	Screens	Vitals	Reports
Customer				
Inventory				
Sales Rep				
Invoices				
Inv Dtl				

Thoroughbred VIP 4.10 06/23/03 13:09:06 tsi

Customer Code 100102
Company Name Computer Inc.
Contact Sue Thompson
Address Route 12
City Madison
State NJ Zip 12441
Phone 201 631-7800
Credit Limit 6010 Discount 34.00
Sales 16628.04
Sales Rep JJ
Comments

The background image for VIP Server Window can be set from within the WorkStation Manager. For more information please see refer to the WorkStation Manager manual (VIP tab).

The title for the VIP Server Window can be set from within the Workstation Manager. For more information please see refer to the WorkStation Manager manual (VIP tab).

The VIP Menu Bar

Many of the VIP options are maintained from within the WorkStation Manager, please refer to that manual. The text displayed in the menu bar can be modified or translated to other spoken languages. Please refer to the Multi-Language for VIP manual

System

Find Current Objects (CTRL-F)
The host controls what graphical object within the main VIP GUI Server window should have the focus. In a Windows environment the user can click on any window displayed on the desktop. This inherent difference between host application code and a windows workstation paradigm can at times create a situation where the current VIP object looses the focus. Use this option to place the focus back on the current VIP object. A Hold Object Focus option in the WorkStation Manager overrides this default behavior. When enabled the user cannot change the focus off the current object. For more information, please refer to the WorkStation Manager Manual (VIP tab).

Thoroughbred Applications (CTRL-P)
Provides access to the hotkey menu (CTL-P). For more information see the Dictionary-IV Administrator Guide (Desk Services/HotKey Menu).

Exit
Terminates the entire Basic session.

Options

Status Bar Position
Allows you to change the position of the VIP status bar. For more information see the WorkStation Manager manual (VIP tab).

Status Bar colors
See the "Message" section below.

View Font
Allows you to set the fonts for the view currently displayed. This is saved on the client and applied when this view is displayed on the client. To set the font for all views, see Views below.

View Colors
Allows you to set the colors for the view currently displayed. This will be saved on the client and applied when this view is displayed. To set the colors for all views, see Views below.

Help

Displays the current version of VIP.

Menus

OPENworkshop Menus

OPENworkshop menus automatically convert to a graphical list. Any change made to the corresponding character definition is reflected in the graphical list. All VIP clients display a graphical list based on the common character definition on the host.

For performance, the graphical list is cached on the client. Each time an OPENworkshop menu is invoked the host compares the last change date and time and the current language code to the graphical list cached on the client. If these do not match, an updated graphical list is generated and cached on the client.

Mouse
A single left mouse click highlights the item, a second left mouse click selects that item, a double left mouse click highlights and selects the menu item.

Scrollbars
Scrollbars are automatically added when needed.

Fonts
The font name, font style, and point size can be controlled from the WorkStation Manager. For more information please see the WorkStation Manager manual (Screen, Menu and Help Fonts tab). Fixed-width fonts are recommended to achieve proper text alignment within the list. Recommended fonts for on-line help are DejaVu Mono, Courier New, and IBMPCDOS. For more information please see the WorkStation Manager manual (Screen, Menu and Help Fonts tab).

Width
The width of the graphical list is determined by menu content or by the menu title. For more information please see the WorkStation Manager manual (VIP tab).

Object Locations
Often with graphical presentations, the desired location of objects is not identical to the character-based location. The graphical menu location can be saved on the client. For more information please see the WorkStation Manager manual (Object Locations tab).

Dictionary-IV Menus

Dictionary-IV menu presentations are screen-based. Graphical Dictionary-IV menus are presented using graphical screen definitions. Graphical Dictionary-IV menus support all the features and functionality of a character-based Dictionary-IV menu and also all the features and functionality of a graphical screen definition.

You can create custom graphical menus using the Graphical Screen Designer. For example, buttons can be added to a graphical menu that:

- Return menu selections to the host.
- Return keystrokes to the host. For example a button can be added that will return a CTL=6 for help.
- Will execute a Windows application. For example, buttons can be added to start a local email client, Excel, Word, or a web browser.

When working with graphical Dictionary-IV menus please refer to Create Graphic Presentation Screens section and the Graphic Screen Designer Section.

Matrix Menus

Matrix menus automatically convert to a grid control. Any change made to the corresponding character definition is reflected in the graphical matrix menu. All VIP clients display a graphical matrix menu based on a common character definition on the host.

For performance, the menu is cached on the client. Each time a matrix menu is invoked the host will compare the last change date and time and the current language code to the graphical menu cached on the client. If these do not match, an updated graphical menu is generated and cached on the client.

Mouse
A single left click selects a cell. When a cell is selected a folder icon will be displayed indicating it is the current active cell. A double left click executes the menu selection defined in the cell.

Fonts
The font name, font style, and point size can be controlled from the WorkStation Manager. For more information please see refer to the WorkStation Manager manual (Screen, Menu and Help Fonts tab). Fixed-width fonts are recommended to achieve proper text alignment within the list. Recommended fonts for matrix menus are DejaVu Mono, Courier New, and IBMPCDOS. For more information please see refer to the WorkStation Manager manual (Screen, Menu and Help Fonts tab).

Object Locations
Often with a graphical presentation, the desired location of objects is not identical to the character based location. The graphical menu location can be saved on the client. For more information please see refer to the WorkStation Manager manual (Object Locations tab).

Selection List

Selection lists automatically convert to a graphical list. These objects are similar to graphical OPENworkshop menus. Because selection lists are generated at run time, and not a Dictionary-IV definition, they are not cached on the client. Each time a list is invoked by the host a graphical selection list is created for the client.

Mouse
A single left mouse click highlights the item, a second left mouse click selects that item, a double left mouse click highlights and selects the menu item. Scrollbars are automatically added when needed.

Scrollbars
Scrollbars are automatically added when needed.

Fonts
Selection lists use the same font settings as OPENworkshop menus.

Width
The width of the graphical list can be determined by menu content or by the menu title. For more information please see refer to the WorkStation Manager manual (VIP tab).

On-line Help

On-line help automatically converts to the appropriate graphical control:

- Windows text control.

- A selection list.

Any change made to the corresponding character definition is reflected in the graphical control. All VIP clients present graphical on-line help based on a common character definition on the host.

For performance, the on-line help is cached on the client. Each time on-line help is invoked the host compares the last change date and time and the current language code to the graphical control cached on the client. If these do not match, an updated graphical control is generated and cached on the client.

Scrollbars
Vertical and horizontal scroll bars are automatically added as needed.

Resize Window
To resize the on-line help window use normal Windows procedure grabbing the left, right, top, bottom or corner to drag the window to the necessary size.

Text Editor

To edit the help text press **F1**. Changes made on the client are updated to the help definition saved on the host. To preserve the column and row format required for a character help definition a fixed font will be used when editing help text within the graphical text control.

With this editor you can place text onto the clipboard for placement in other applications. You can also place text from other applications onto the clipboard, and then place them in the text editor window.

Fonts

The font name, font style, and point size is controlled from the WorkStation Manager. Fixed-width fonts are recommended to achieve proper text alignment. Help modules containing business graphic characters must use a font that supports those characters. Recommended fonts for on-line help are DejaVu Mono, Courier New, and IBMPCDOS. For more information please see refer to the WorkStation Manager manual (Screen, Menu and Help Fonts tab).

Location of Objects

Often with a graphical presentation, the desired location of objects is not identical to the character-based location. The graphical help location can be saved on the client. For more information please see refer to the WorkStation Manager manual (Object Locations tab).

Text Field

Text fields automatically convert to a graphical text control. These objects are similar to graphical on-line help. Text fields are not cached on the client. Each time a text field is invoked by the host a graphical text control is created for the client.

Changes made on the client are updated to the data files on the host. To preserve the column and row format required for a character text, a fixed font is used when editing text fields within the graphical text control.

With this editor you can place text onto the clipboard for placement in other applications. You can also place text from other applications onto the clipboard, and then place them in the text editor window.

Text fields use the same font settings as on-line Help.

Messages

Messages can be displayed in the VIP Status Bar control or using a message dialog box. To select the presentation style, see the WorkStation Manager manual, Options (VIP tab, Status Bar and Message Objects).

Displaying Messages

Status Bar
All Prompt/Constant, View Prompt, and Screen Prompt messages are always displayed in the VIP Status Bar control.

NOTE: Prompt messages by definition do not require user action. Dialog box controls require user action.

The status bar contains the message text, buttons and an input control for input messages.

The location of the status bar can be set from within the WorkStation Manager. For more information see the WorkStation Manager manual (VIP tab).

Message colors displayed in the Status Bar can be set from the VIP Menu Bar. From Options select Status Bar Colors. From here the following color properties for Status Bar Messages can be set:

- Button Text.
- Prompt Msg and NonInput Msg Text.
- Input Msg, Yes/No Msg and Active Button Text.

The Status Bar message font and point size can be set from within the WorkStation Manager. For more information see the WorkStation Manager Manual (Screens, Menus and Help Fonts tab).

Dialog Boxes
Non-input, yes/no and input messages can be displayed in the Status Bar or in a message Dialog Box. The dialog box will contain the message text, buttons and an input control for input messages. Dialog boxes are displayed in the center of the VIP Server Window.

Message Types

Prompt/Constant
These messages always display in the status bar. Prompt messages can be displayed as plain text or can be converted to buttons. For more information see the WorkStation Manager Manual (VIP tab).

Non-input
VIP automatically adds an Ok, Help and Cancel button to these message types.

- Ok will return <CR> to the host.
- Help will return <F6> to the host.
- Cancel will return <F4> to the host.

Yes/No
VIP automatically adds a Yes, No, Help, and Cancel button to these message types.

- Yes returns the current language "Yes" value to the host. This response is processed as a single character input (INPUT SIZ=1). For more information see the Dictionary-IV Administrator Guide (Multi-Language).

- No returns the current language 'No' value to the host. This response is processed as a single character input (INPUT SIZ=1). For more information see the Dictionary-IV Administrator Guide (Multi-Language).

- Help will return <F6> to the host.

- Cancel will return <F4> to the host.

Input
VIP automatically adds an input control, an Ok, Help, and a Cancel button to these message types. For messages with valid value lists defined VIP automatically adds a button for each element in the valid value list.

- Ok returns <CR> to the host.

- Help reruns <F6> to the host.

- Cancel returns <F4> to the host.

- For messages with valid value lists defined, an invalid response from the operator will display a graphical list of valid values for selection.

Select
VIP automatically adds a button for each element in the select list. In addition it adds an OK, Help, and Cancel button.

- Clicking on a selection button returns that selection value to the host.

- Ok returns <CR> to the host.

- Help reruns <F6> to the host.

- Cancel returns <F4> to the host.

Views

OPENworkshop and Dictionary-IV views automatically convert to a grid control. Any change made to the corresponding character definition is reflected in the grid control. All VIP clients present a graphical view based on a common character definition on the host.

For performance, the graphical view is cached on the client. Each time a view is invoked the host compares the last change date and time to the graphical view cached on the client. If these do not match, an updated graphical view is generated and cached on the client.

View Prompt Messages

View prompt messages are displayed in the VIP Status Bar control. The Status Bar control is refreshed when moving from cell to cell. Please see the Messages section earlier in this chapter.

Location of Objects

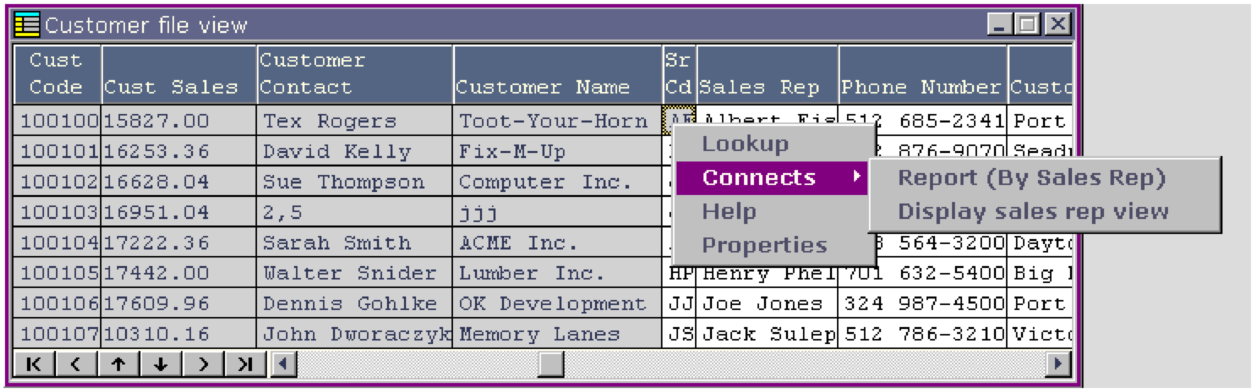
Often with a graphical presentation, the desired location of objects is not identical to the character based location. The graphical view location can be saved on the client. For more information please see refer to the WorkStation Manager manual (Object Locations tab).

Mouse Control

A single left click in a cell will select a cell displaying it in an alternate color. The cell displays in the color set in the "Lock Colors" definitions. For more information see the WorkStation Manager manual (View Fonts and Colors tab).

NOTE: If the current cell is within a locked column, then the current cell is displayed using the default color properties on that tab.

Once a cell is selected a right mouse click displays a context menu. This menu displays options based on the attributes of the data element as defined by the format definition.



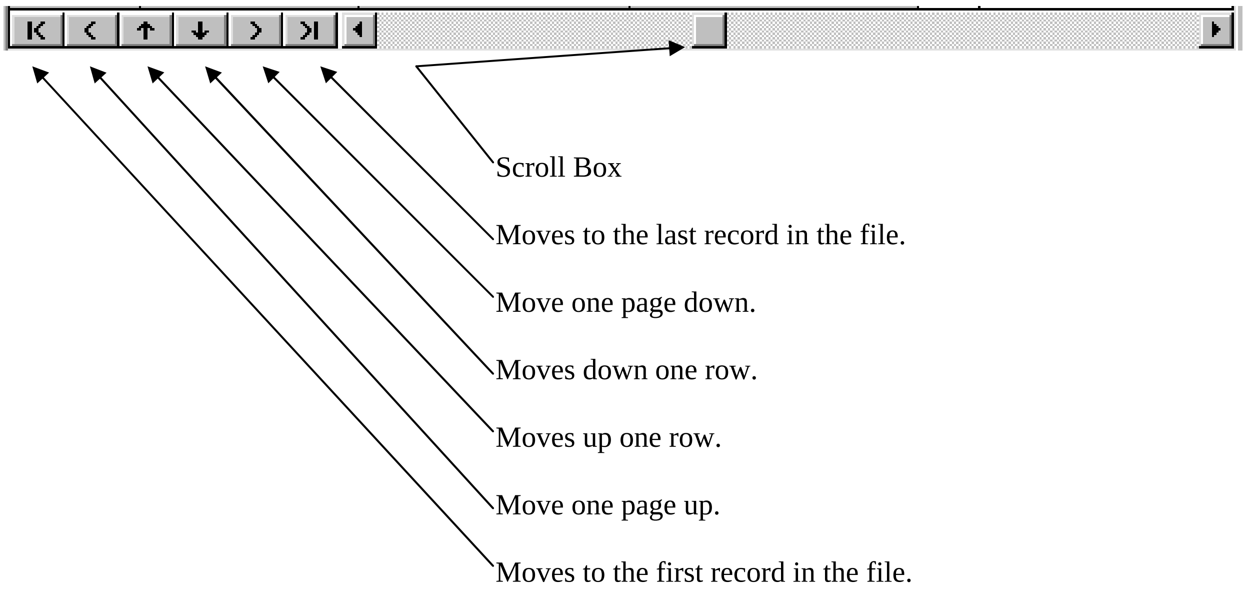
Cust Code	Cust Sales	Customer Contact	Customer Name	Sr Cd	Sales Rep	Phone Number	Custo
100100	15827.00	Tex Rogers	Toot-Your-Horn	AF	Albert Fis	512 685-2341	Port
100101	16253.36	David Kelly	Fix-M-Up			876-9070	Seadı
100102	16628.04	Sue Thompson	Computer Inc.				
100103	16951.04	2,5	jjj				
100104	17222.36	Sarah Smith	ACME Inc.			564-3200	Dayto
100105	17442.00	Walter Snider	Lumber Inc.	HP	Henry Phel	701 632-5400	Big I
100106	17609.96	Dennis Gohlke	OK Development	JJ	Joe Jones	324 987-4500	Port
100107	10310.16	John Dworaczyk	Memory Lanes	JS	Jack Sulep	512 786-3210	Victo

- **LookUp:** If a LookUp is defined for the data element, select the option from the context menu to invoke the LookUp. If a LookUp is not defined, the option is disabled.
- **CONNECTs:** If one or more CONNECTs are defined for the data element, selecting one of the connect options from the context menu invokes the CONNECT. If CONNECTs are not defined, this option is disabled.
- **Help:** If help is defined for the data element, selecting this option from the context menu invokes Help. If help is not defined for the data element, this option is disabled.
- **Properties:** This is only available in developer mode. Selecting this option displays the data element attributes. This is equivalent to pressing **F11** in character mode.

A double left click of the selected cell activates that field for editing. A cell activated for editing displays using the Desktop Display Properties for a selected field indicating it can be edited. Clicking off an activated field will return a <CR> to the host.

By default data is aligned within a cell using the same rules as a character view. Data is centered in a column when the column width is greater than the data width. Masking data to the defined column width results in data being either left or right justified. In a graphical presentation this does not always yield the desired results. Optionally, you can set VIP to right align all date fields, numeric fields and all fields with padding types of 1 or 3. All other field types and padding types are left justified. This is controlled from within the WorkStation Manager. For more information see the WorkStation Manager (VIP tab).

The grid control contains a scroll bar with buttons to navigate within the view. The scroll bar buttons work as follows:



Scroll Box

Moves to the last record in the file.

Move one page down.

Moves down one row.

Moves up one row.

Move one page up.

Moves to the first record in the file.

Resizing the Grid Control

The grid width and grid column widths are expandable using standard Windows procedures. Grab the window edges or corners to change the grid width. Grab the column edges or corners to change the column width.

The new grid width and grid column widths are always automatically saved on the workstation. This feature is separate from the option to save View locations.

Selecting Font and Color

Default fonts and colors can be set from the WorkStation Manager. For proper cell alignment fixed-width fonts are recommended. Default colors can be set for the view grid, column headings, and locked columns. See the WorkStation Manager Options (View Font and Colors tab). Recommended fonts for on-line help are DejaVu Mono, Courier New, and IBMPCDOS. For more information please see refer to the WorkStation Manager manual (View Font and Colors tab).

Custom view fonts and colors can be set for individual views. When a view displays, from the VIP Menu Bar select **Options → View Fonts**. This font selection will only be applied to the current view. From the VIP Menu Bar select **Options → View Colors** to select colors. These color selections will only be applied to the current view. To save the view settings, you must have selected **Yes** for View Location in WorkStation Manager Options (Object Locations tab).

Dynamic View Colors

You can use application code to control view colors at runtime. For example, it is possible to have all negative numbers in a column to be displayed using white text on a red background. Background and foreground colors can be set for a cell, a column, a range of columns, a row or a range of rows. This is accomplished by either passing SETCOLOR syntax or the name of a View Color Method to the View class.

SETCOLOR

The SETCOLOR syntax is user friendly but does have some limitations. It does not support user defined columns and does not support compound conditionals within a single WHEN clause. This approach to setting view colors is preferred when the color rules are fairly simple and straightforward.

The SETCOLOR rules are passed to the View class in VIEW$[29].

SETCOLOR SYNTAX

`SETCOLOR` ***`foreground+background`*** **`BY COL.`*****`data-name`*****`|ROW WHEN`** ***`condition`***

SETCOLOR Required syntax.

foreground Required color word to define the foreground color. Use any valid Basic color keyword. For example: WHITE, BLACK, etc.

NOTE: The + is required syntax used to separate foreground color from background color.

background Required color word to define the background color. Use any valid Basic color keyword. For example: MAGENTA, LBLUE, etc.

BY Required syntax.

COL.*data-name* To set the color for a column, specify the keyword COL before entering the column data-name. For example, COL.CUST-CODE. The data elements supplied with the COL option must be defined in the primary Format definition. For example the OEVCUST view can only reference data elements defined in OEFCUST. You cannot refer to a user-defined column (ex: OELSLRP.SR-NAME) using the simple SETCOLOR syntax. The COL option and the ROW option are mutually exclusive within the same rule.

ROW To set the color for the entire row use the ROW option.

WHEN Required syntax.

condition Supply a SELECT WHEN style condition. The same rules that apply to passing a SELECT WHEN clause to the View class apply to the SETCOLOR WHEN clause. This is done in VIEW$[29]. For more information see the OPENworkshop manual.

To set multiple rules separate each by a comma (,). Each rule is evaluated in the order in which it is received and the number of rules is only limited by the maximum string size (64k).

SETCOLOR rules are applied to the row when either moving off a column or moving off the row.

SETCOLOR Examples

The following example will set an entire row to white on magenta when the CUST-CODE is 100102.

```
DIM VT$[29];
VT$[1]="OEVCUST",
VT$[29]="SETCOLOR WHITE+MAGENTA BY ROW WHEN" +
         "#OEFCUST.CUST-CODE = "+QUO+"100102"+QUO;
CALL "OO3A",VT$[ALL]
```

The following example will set CUST-SALES column to white on cyan when the CUST-SALES are greater than 16,999.00."

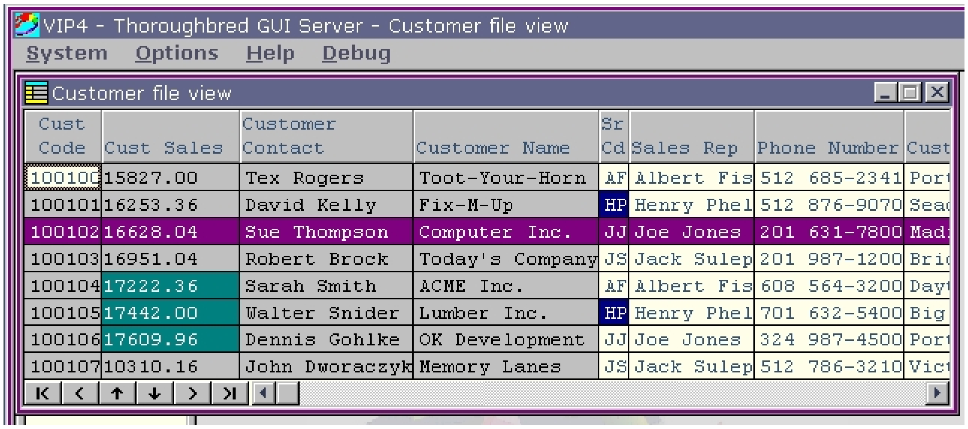
```
DIM VT$[29];
VT$[1]="OEVCUST",
VT$[29]="SETCOLOR WHITE+CYAN BY COL.CUST-SALES WHEN"
         "#OEFCUST.CUST-SALES=> 16999";
CALL "OO3A",VT$[ALL]
```

The following example will set the SR-CODE column to white on blue when the SR-CODE = "HP."

```
DIM VT$[29];
VT$[1]="OEVCUST",
VT$[29]="SETCOLOR WHITE+CYAN BY COL.SR-CODE WHEN"+
         "#OEFCUST.SR-CODE = "+QUO+"HP"+QUO;
CALL "OO3A",VT$[ALL]
```

The following will apply all three rules in sequence:

```
DIM VT$[29];
VT$[1]="OEVCUST",
VT$[29]="SETCOLOR WHITE+MAGENTA BY ROW WHEN" +
        " #OEFCUST.CUST-CODE = "+QUO+"100102"+QUO +
        "; SETCOLOR WHITE+CYAN BY COL.CUST-SALES WHEN"+
        " #OEFCUST.CUST-SALES=> 16999"+
        "; SETCOLOR WHITE+CYAN BY COL.SR-CODE WHEN"+
        " #OEFCUST.SR-CODE = "+QUO+"HP"+QUO;
CALL "OO3A",VT$[ALL]
```



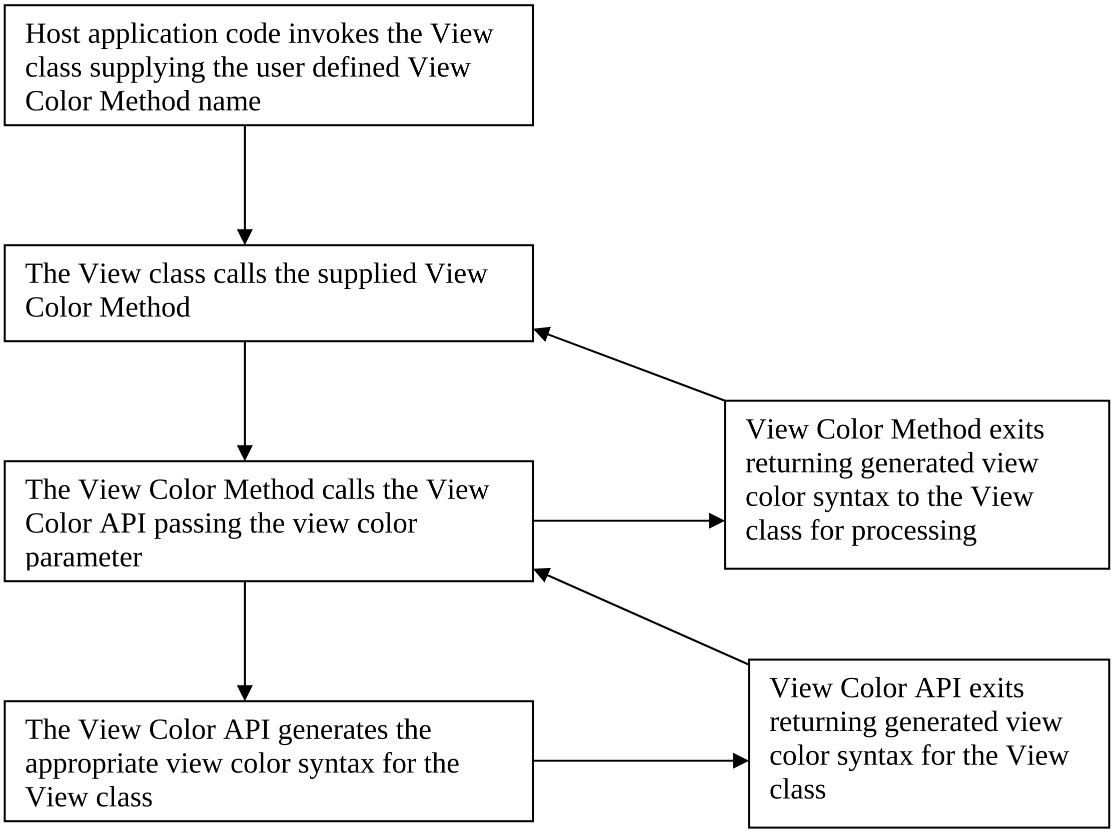
Cust Code	Cust Sales	Customer Contact	Customer Name	Sr Cd	Sales Rep	Phone Number	Cust
100100	15827.00	Tex Rogers	Toot-Your-Horn	AF	Albert Fis	512 685-2341	Port
100101	16253.36	David Kelly	Fix-M-Up	HP	Henry Phel	512 876-9070	Sea
100102	16628.04	Sue Thompson	Computer Inc.	JJ	Joe Jones	201 631-7800	Mad
100103	16951.04	Robert Brock	Today's Company	JS	Jack Sulep	201 987-1200	Bri
100104	17222.36	Sarah Smith	ACME Inc.	AF	Albert Fis	608 564-3200	Day
100105	17442.00	Walter Snider	Lumber Inc.	HP	Henry Phel	701 632-5400	Big
100106	17609.96	Dennis Gohlke	OK Development	JJ	Joe Jones	324 987-4500	Port
100107	10310.16	John Dworaczyk	Memory Lanes	JS	Jack Sulep	512 786-3210	Vict

View Color Method

A user defined View Color Method does require application code but does not have the same limitations found in the simple SETCOLOR syntax. This approach for setting view colors is preferred when the color rules are complex and more flexibility is required.

The user defined View Color Method name is passed to the View class when performing a CONNECT VIEW. The View class will execute the View Color Method when moving off a column or moving off a row. The View Color Method then constructs the appropriate color parameters to be applied to the view by the View class. The View Color Method must pass the color parameters to the View Color API for parsing. The View Color API will interpret the supplied color parameters and generate compact syntax that is in turn processed by the View class.

The following diagram illustrates the process flow when using a View Color Method:


Host application code invokes the View class supplying the user defined View Color Method name
The View class calls the supplied View Color Method
The View Color Method calls the View Color API passing the view color parameter
The View Color API generates the appropriate view color syntax for the View class
View Color Method exits returning generated view color syntax to the View class for processing
View Color API exits returning generated view color syntax for the View class

View Class SetColor Method Name Syntax

SETCOLOR METHOD=*method-name*

SETCOLOR METHOD Required syntax.

method-name The name of the method to be executed. Pass the View class the View Color Method name in VIEW$[29]. For more information about VIEW$[29] see the OPENworkshop manual.

View Color Method CALL/ENTER List

```
METHOD V$[ALL],VCOLR$[ALL],LNK$[ALL]
```

```
V$[x,y] View Class Array Supplied by the View Class
```

[0,0]	(1,1)	Length of column attribute entry.
	(2,1)	Number of view used column.
	(3,1)	Number of view deleted columns.
	(4,1)	Number of view heading rows.

	(5,n)	New column id and column number string.
	(6,1)	New column number.
	(7,1)	Next new column id.
	(8,1)	Next new column number.
	(9,n)	n column id & n column number.
[0,1]	(1,1)	Column window address (column).
	(2,1)	Column window address (row).
	(3,1)	Column width.
	(4,1)	Column dataname number. 00 implies new column.
	(5,1)	New column id.
	(6,n)	New column data (build by view method). V$[0,n](6) will be displayed by the view CLASS when the next row is printed. V$[0,n] Same as V$[0,1]

VCOLR$[ALL] View Color Array

[0] API returns "." when no errors else returns one of the following error codes:

"100"	Not enabled
"101"	Supplied parameters invalid
"102"	Supplied function requires supplied color parameters
"103"	Supplied row and column parameters invalid
"104"	Supplied row range invalid
"105"	Supplied column range invalid
"106"	Supplied cell range invalid
"107"	Supplied foreground color invalid
"108"	Supplied background color invalid

The content of [0] is used by the view class to determine how it should process colors. Upon returning from the view color method:

If VCOLR$[0]="." AND VCOLR$[3]<>"" the view class will assume a valid color command string was built by the API and will apply it.

If the above conditions fail the view class will build a command string for this row to apply the default view colors.

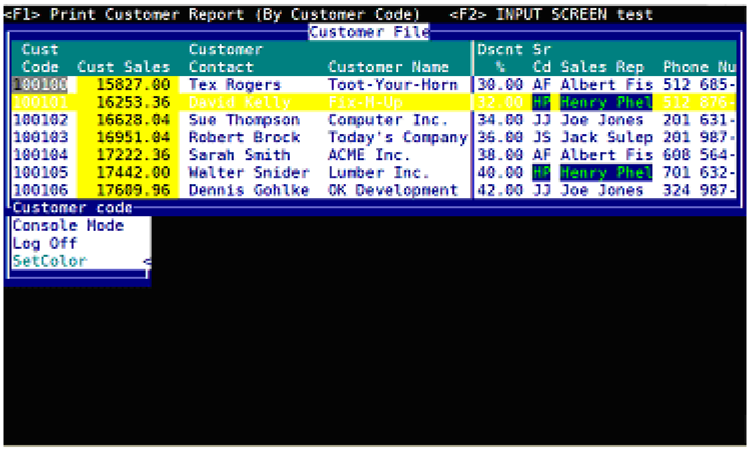
[1] Column, row information supplied by the View class

(1,3)	Current column number, 1 based.
(4,3)	Current row number, 1 based.
(7,3)	First data row number, 0 based.
(10,3)	Last data row number, 0 based.

[2] Color parameters built by the View Color Method, passed to the color API.

(1,1)	Mode: "S" Set color parameters supplied in VCOLR$[2] "R" Reset entire view back to default colors
(2,3)	Starting column number, 1 based. "999" to change all columns in the starting row (5,3). "998" for the current column in the view. When VCOLR$[2](1,1)=”S” color parameters in bytes 5 through 61 must be supplied. Multiple color parameters are supported. When supplying multiple color parameters, each set must be padded to 30 bytes.
(5,3)	Starting row number, 0 based and include heading rows, must be >=VCOLR$[1](7,3) and <=VCOLR$[1](10,3). “999” to change all rows for the column specified in (2,3). "998" to change the current row.
(8,3)	Ending column to define a range. "999" to change all columns in the ending row. " " no ending column.
(11,3)	Ending row to define a range must, be >=VCOLR$[1](7,3) and <=VCOLR$[1](10,3). "999" to change all rows in the ending col. " " no ending row. **NOTE:** When both ending column and ending row are " " the starting column and row values are applied to the ending column and row values.
(14,9)	Foreground color, Basic color keyword padded 9 bytes
(23,9)	Background color, Basic color keyword padded 9 bytes
(32,30)	Next set of color parameters

NOTE: Character mode does not support RGB color notation, only Basic keywords.



```
<F1> Print Customer Report (By Customer Code)    <F2> INPUT SCREEN test
                                Customer File
 Cust                Customer                         Dscnt Sr
 Code   Cust Sales   Contact         Customer Name      %   Cd Sales Rep   Phone Nu
100100    15827.00   Tex Rogers      Toot-Your-Horn    30.00 AF Albert Fis 512 685-
100101    16253.36   David Kelly     Fix-M-Up          32.00 HP Henry Phel 512 876-
100102    16628.04   Sue Thompson    Computer Inc.     34.00 JJ Joe Jones  201 631-
100103    16951.04   Robert Brock    Today's Company   36.00 JS Jack Sulep 201 987-
100104    17222.36   Sarah Smith     ACME Inc.         38.00 AF Albert Fis 608 564-
100105    17442.00   Walter Snider   Lumber Inc.       40.00 HP Henry Phel 701 632-
100106    17609.96   Dennis Gohlke   OK Development    42.00 JJ Joe Jones  324 987-
Customer code
```

The format is:

#rrggbb

where:

# is required syntax

rr is a red value

gg is a green value

bb is a blue value

For example: #ff00ff

When neither a foreground nor a background color is supplied the view default colors are applied.

VCOLR$[3] SetColor command string built by the API and returned to the View Color Method. This string must be returned to the View class.

LNK$[ALL] - See link array in the OPENworkshop Manual.

Example Call View Class

```
METHOD MSGX$[ALL]
```

Call View Class with a View Color Method name.

```
....
PROCEDURE

      DIM VT$[29];
      VT$[1]="OEVCUST",
      VT$[29]="SETCOLOR METHOD=OEVCOLOR";
      CALL "OO3A",VT$[ALL];
      GOTO CUEXIT
```

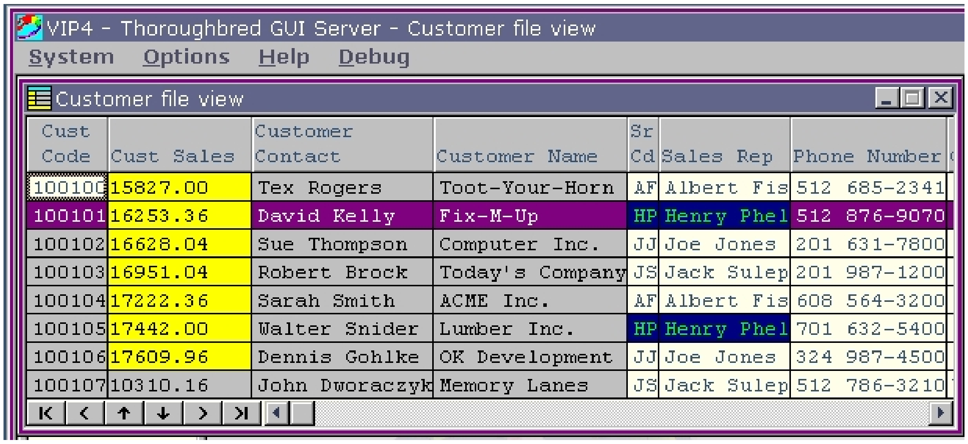
Example View Color Method (OEVCOLOR):

```
METHOD V$[ALL], VCOLR$[ALL],LNK$[ALL]
```

This method (OEVCOLOR) is called by the View Class.

```
....
PROCEDURE
FORMAT INCLUDE #OEFCUST,OPT="NONE";

! By default if VCOLR$[3] does not return to the view class a color command.
!    string, the current row will be initialized to default view colors. It
!    is not necessary to reset row colors back to defaults.


! When CUST-CODE is 100101 display entire row using white on magenta:

      IF #OEFCUST.CUST-CODE = "100101" ! If row is customer code 100101
         VCOLR$[2]=:+                  !    Set color parms:
             "999"+                    !         all cols in this row +
             VCOLR$[1](4,3)+           !         this row number +
             "   "+                    !         no ending col (range) +
             "   "+                    !         no ending row (range) +
             PAD("WHITE",9)+           !         foreground color +
             PAD("MAGENTA",9)          !         background color
      FI;                              ! enidf
```

```
! The next examples perform a test on a column basis, regardless of the
! current column. This code is an example of how to determine which column
! in a view contains what data element name. In most real-life cases, this
! is not necessary, the format entry number in the view array is sufficient.
! This is simply an example if this type of logic should be required.


! When CUST-SALES > 15,000 display that column in this row using
!   black on yellow

      IF #OEFCUST.CUST-SALES > 15000   ! If cust sales > 15000
         FX$="#OEFCUST";               !    Set format name
         FOR V=1 TO NEA("V$",2);       !    Loop through data elements
            EN=DEC(V$[0,V](4,1)),      !      Get format entry nbr
            DN$=ATR(FX$,EN,21);        !      Get its data name
            IF CVT(DN$,128)="CUST-SALES"!      If its the right one
               VCOLR$[2]=:+            !          Set color parms:
                   STR(V-1:"000")+     !               this col +
                   VCOLR$[1](4,3)+     !               this row +
                   "   "+              !               no ending col (range) +
                   "   "+              !               no ending row (range) +
                   PAD("BLACK",9)+     !               foreground color +
                   PAD("YELLOW",9),    !               background color
               V=NEA("V$",2)+1         !         Force end of loop
            FI;                        !      endif
         NEXT V                        !    Next data element in format
      FI;                              ! endif

! When SR-CODE=HP print both the SR-CODE and sales rep name (<Ax>) using
! light green on blue. This uses a column range to set the color of two
! consecutive columns.

      IF #OEFCUST.SR-CODE="HP"         ! If sales rep is H
         FX$="#OEFCUST";               !    Set format name
         FOR V=1 TO NEA("V$",2);       !    Loop through data elements
            EN=DEC(V$[0,V](4,1)),      !      Get format entry nbr
            DN$=ATR(FX$,EN,21);        !      Get its data name
            IF CVT(DN$,128)="SR-CODE"  !      If its the right one
               V0$=V$[0,0],            !         Get col attr string
               EL=ASC(V0$(1,1)),       !         Get len of col attr entry
               NC$=V0$(EL),            !         Get 'new col A<x>' string
               NC=POS("A"=NC$,2);      !         Scan for new col "A"
               IF NC                   !         If found rep name
                  CN=ASC(NC$(NC+1,1))  !            Get rep name col nbr
               FI;                     !         endif
               VCOLR$[2]=:+            !         Set color parms:
                   STR(V-1:"000")+     !              this col +
                   VCOLR$[1](4,3)+     !              this row +
                   STR(CN-1:"000")+    !              ending col (name) +
                   VCOLR$[1](4,3)+     !              this row +
                   PAD("LGREEN",9)+    !              foreground color +
                   PAD("BLUE",9),      !              background color
               V=NEA("V$",2)+1         !         Force end of loop
            FI;                        !      endif
         NEXT V                        !   Next column
      FI;                              ! endif

* Call API to construct the set color command string required by view class

      IF LEN(VCOLR$[2])>1              ! If have something
         VCOLR$[2]="S"+VCOLR$[2];      !    Set API function: set colors
         CALL "OO3AV01",VCOLR$[ALL],   !    Call API, color array,
                          V$[ALL]      !              view array
      FI;                              ! endif

* Will exit back to view processing with VCOLR$[3] built for the view class.

      GOTO CUEXIT
```


VIP4 - Thoroughbred GUI Server - Customer file view
System Options Help Debug
Customer file view

Cust Code	Cust Sales	Customer Contact	Customer Name	Sr Cd	Sales Rep	Phone Number
100100	15827.00	Tex Rogers	Toot-Your-Horn	AF	Albert Fis	512 685-2341
100101	16253.36	David Kelly	Fix-M-Up	HP	Henry Phel	512 876-9070
100102	16628.04	Sue Thompson	Computer Inc.	JJ	Joe Jones	201 631-7800
100103	16951.04	Robert Brock	Today's Company	JS	Jack Sulep	201 987-1200
100104	17222.36	Sarah Smith	ACME Inc.	AF	Albert Fis	608 564-3200
100105	17442.00	Walter Snider	Lumber Inc.	HP	Henry Phel	701 632-5400
100106	17609.96	Dennis Gohlke	OK Development	JJ	Joe Jones	324 987-4500
100107	10310.16	John Dworaczyk	Memory Lanes	JS	Jack Sulep	512 786-3210

Screens - Introduction

Except where otherwise noted, this section also applies to Dictionary-IV graphical menus.

Unique to screens is the ability to use default or custom graphical screens. For both graphical screen types, any change made to the corresponding character screen definition will be reflected in the graphical screen. All VIP clients will present a graphical screen based on a common character definition on the host. For better performance, the graphical screen is cached on the client. Each time a screen is invoked the host will compare the last change date and time and language code to the graphical screen cached on the client. If these do not match and a custom screen exists on the host, the custom screen will be downloaded and cached on the client. If these do not match and a custom screen does not exist on the host, a new default screen will be generated and cached on the client.

Control Types

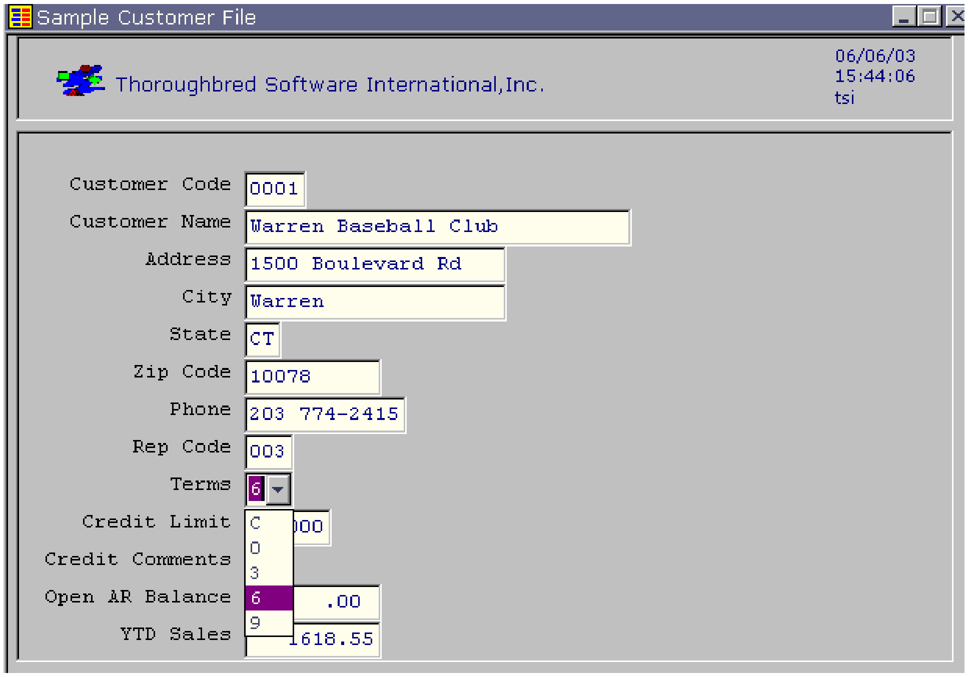
Both default and custom screens will create control types based on the data element attributes as defined by the format.

- Standard input fields will be collected using a text box control.
- Calculated (formula) fields will be displayed using a text box control using special foreground and background colors. See Special Field Colors later in this section.
- Security fields will be displayed using a text box control using special foreground and background colors. See Special Field Colors later in this section.
- Text fields will be displayed and edited using a graphical text editing control.
- Valid Value lists will be displayed and selections collected using a ComboBox control. Clicking on the ComboBox button will display a drop down list containing a list of valid values.

- Context Menus are automatically generate for all fields. A right mouse click on the active field will display a context menu. This menu displays options based on the attributes of the data element as defined by the format definition. For more information on data element Context Menus see the Views section of Creating Graphical Presentations.

In addition to the above, custom screens support button controls and image controls. Please see the Screens-Custom in a following section.

The following image is an example of a default graphical screen with a ComboBox control used for processing the Valid Value list defined for the Terms field.


Sample Customer File

Thoroughbred Software International,Inc.

06/06/03
15:44:06
tsi

Customer Code 0001
Customer Name Warren Baseball Club
Address 1500 Boulevard Rd
City Warren
State CT
Zip Code 10078
Phone 203 774-2415
Rep Code 003
Terms 6
Credit Limit 00
Credit Comments
Open AR Balance .00
YTD Sales 1618.55

Special Field Colors

Colors for calculated (formula) fields, security fields, and highlighted (current) field can be set in the WorkStation Manager. These color properties can be applied at run time to both default screens and custom screens. For more information see the WorkStation Manager manual (Special Fields Color tab).

NOTE: This does not apply to Dictionary-IV graphical menus.

Screen Location

Often with a graphical presentation, the desired location of objects is not identical to the character based location. The graphical screen location can be saved on the client. For more information please see refer to the WorkStation Manager manual (Object Locations tab).

Screen Messages

Screen prompt messages will be displayed in the VIP Status Bar control. The Status Bar control is refreshed when moving from field to field. Non-input, Yes/No, and Input messages can be displayed in either the Status Bar control or in a Message Dialog box. Please see the Messages section earlier in this chapter.

NOTE: This does not apply to Dictionary-IV graphical menus.

Default Screens

When a screen definition is invoked, Dictionary-IV will determine if a custom screen definition exists. If not, a default screen will be generated based on the character screen definition. Default screens are cached on the client. Unlike custom screens they are not saved on the host.

Several tools are available to enhance the interpretation of a character screen into a graphical object.

- The font for default screens can be set in the workstation manager. For more information see the WorkStation Manager (Screens, Menu, and Help Fonts tab).

- Use Screen filters. For more information see the Screen Filters section later in this manual.

- Default screens also support the use of screen templates. Optionally, when a default screen is generated a screen template will be applied. Please refer to the Templates section below.

- Background and foreground colors can be defined for special field types. For more information see the WorkStation Manager manual (Special Field Colors tab).

The combination of special field colors, screen filters and templates provide a stylized graphical screen without the need to create custom graphical screens.

Custom Graphical Screens

Custom graphical screens are highly customized default screens that are saved on the host in the GUI Dictionary (IDDBB) providing a common graphical screen to all VIP clients.

NOTE: If you are upgrading from earlier versions of VIP where screens were not saved on the host, please see the GUI screen import utility later in this manual.

The custom graphical screen definition is an extension of a character screen; it must have a corresponding character screen definition. The two definitions share a common screen header saved in Dictionary-IV (IDDBD). The graphical screen details are saved in the GUI Dictionary (IDDBB).

Graphical screens are fully integrated with Dictionary-IV maintenance and Dictionary-IV utilities. For example the Dictionary-IV Copy/Rename/Delete functions will be performed on both the selected character screen and the corresponding graphical screens. The Backup/Restore utilities will include both the character screen definition and the corresponding graphical screen details.

Custom graphic screens are automatically updated to reflect changes made to the corresponding character screen and the format definition. The character screen definitions and the graphical screen definitions are automatically kept in sync. This eliminates the need to regenerate a new default screen when either the format or the character screen is modified. Depending upon the nature of the changes, it may be necessary to make some edits to the graphical screen definition in order to achieve the desired presentation. The following describes how these changes impact the look of a graphical screen definition.

Adding New Fields

When a new field is added to a character screen definition, that field is automatically appended to the corresponding custom graphical screen definition. This is true for both input fields and formula fields.

NOTE: Adding a new field to a format definition does not impact either the character or the custom graphical screen definition.

Control Types

The control type is based on the format attributes of the field. Both the input control and the corresponding label (field description) will be added. For more information on control types, please see the beginning of this Screens section.

Labels

If a data element description exists, it is used for the corresponding label control. If not, the data element name is used. For formulas, an attempt will be made to determine what text to the left of the formula is the field description. Going from left to right it will scan for the first instance of two or more spaces. It will assume all text from that point to the formula field should be used for the label control. If it fails to determine the label text using this rule, the label control will default to the text: "*FORMULA*". This text can be modified using the Graphic Screen Designer.

Properties

Screen maintenance will analyze the custom graphical screen definition and determine by control type, the most commonly used font and color properties. Those properties will be used when adding the new field. If common properties cannot be determined, default properties will be applied.

Size

The size of a control is determined based on its font properties. The size of a character determined by font face (typeface), style, and point size is used to create the control height and width. When an input control is added, the width of the control is calculated by multiplying the average size of a character in pixels, times the input size. The width of a label control is calculated using the length of the text. The average character size in pixels (font mapping) is maintained in a help module OOVFONT. For more information on maintaining OOVFONT please refer to the Font Mapping section in this document.

Location

When a new control is added, the control is simply appended to the end of the GUI screen definition. If necessary, the height of the screen is adjusted accordingly. It is nearly impossible to determine where a control should be placed. The use of various fonts, panels, images, buttons, etc. result in a GUI presentation that is very different from the character presentation. In addition GUI screens can be arranged using a column structure: top down, left to right. Using a row structure: left to right, top down, or it can be arbitrary: a combination of column and row structures.

Deleting Fields

When a field is deleted from either the character screen or the format definition, that field is automatically removed from the corresponding custom graphical screen definitions.

Screen Maintenance: When a field or formula is deleted from a character screen, that field or formula control is removed from the corresponding custom graphical screen definition.

Format Maintenance: When a field is deleted from a format definition, all custom graphical screen definitions linked to that format are automatically updated; removing the field from the custom graphical screen.

Because there is no link to cross-reference an input control and the corresponding label control, only the input control is removed from the custom graphical screen.

Changing Field Attributes

When the attributes of a data element are changed in format maintenance, those changes are reflected in all custom graphical screens linked to that format definition. The following are some examples of how a custom graphical screen might be impacted:

- If the size of a data element is changed, the maximum input size will be updated.

- If a valid value is added to a data element, custom graphical screens will be updated to use a List Box control for that field at run time.

- If the format entry number is changed, custom graphical screens will be updated to reference the correct format entry number.

- If the data element name is changed, custom graphical screens will be updated to reference the correct data element name.

- If either the data element name or data description is changed, an attempt to identify and update the corresponding Label control is made. If an exact match on the old data element or the old data description can be made, the Label control text property will be updated with the new data element name or description.

Changing Element Order

When the screen collection order is changed form within screen maintenance, those changes will be reflected in the corresponding custom graphical screen definition. There is no impact to the custom graphical screen presentation.

Copy, Delete, and Rename Definitions

Screen, format, link and library maintenance will include the custom graphical screen definitions when a copy, delete or rename is done. For example:

Copying a character screen definition will also copy the corresponding custom graphical screen definition.

Deleting a character screen definition will display a prompt message allowing you to delete both the character screen and the custom graphical screen or just the custom graphical screen.

NOTE: It is not possible to delete just the character screen; the custom graphical screen is a component of the character screen definition.

Renaming a character screen definition will also rename the corresponding custom graphical screen definition.

Renaming a format definition will search all custom graphical screen definitions and automatically adjust the format definition name saved with the custom graphical screen.

Backup/Restore

The Backup and Restore utilities will include custom graphical screen definitions and screen templates (see Templates below). When screens are selected for backup and a custom graphical screen exists, a second backup file will be created. The backup will append "VIP" to the end of the standard backup file name. For example a backup file IDDbkTSI will result in a second backup file named IDDbkTSIVIP containing the custom graphical screen definition details.

NOTE: When moving or copying a backup file from one directory to another be sure to move/copy both backup files.

When renaming a backup file be sure to rename both backup files.

When erasing a backup file be sure to erase both backup files.

The custom graphical backup file will only be created when screen definitions have been selected and corresponding custom graphical screen definitions exist.

When a custom graphical backup file is created, it will include the same time stamp record as the standard backup file. These matching timestamps are used to determine if the two backup files are in sync.

The restore operation will compare the time stamp of both backup files. If these time stamps do not match, a warning message will be displayed.

The restore operation will attempt to open a corresponding custom graphical backup file using the same backup file name rule (e.g. IDDbkTSI, IDDbkTSIVIP). If the custom graphical backup file is found, the custom graphical screens will be restored. If it is not found, a warning message will be displayed.

Both the backup "Edit" and "Refresh" options support custom graphical screens. If appropriate, the custom graphical backup file will be created and the time stamps updated.

Templates

Screen templates are specialized graphical screen definitions used to assign visual properties and characteristics to graphical screens and Dictionary-IV graphical menus. Templates can be applied to default screens at runtime or they can be applied to a graphical screen loaded in the Graphic Screen Designer.

Because screen templates are saved on the host in the Dictionary, they require a corresponding character screen definition. Like other graphical screens, the corresponding character screen definition is stored in Dictionary-IV (IDDBD). The graphical details are stored in the GUI Dictionary (IDDBB).

Following standard Dictionary-IV naming conventions, on the host a template name is from 3 to 8 characters where the first two characters are the library name. A screen template must reference the format definition UTFVT. This is a standard format that must be used by all template definitions. The GUI screen type for a template is a type 4. A default template, UTSV001 is supplied and installed both on the host and the client.

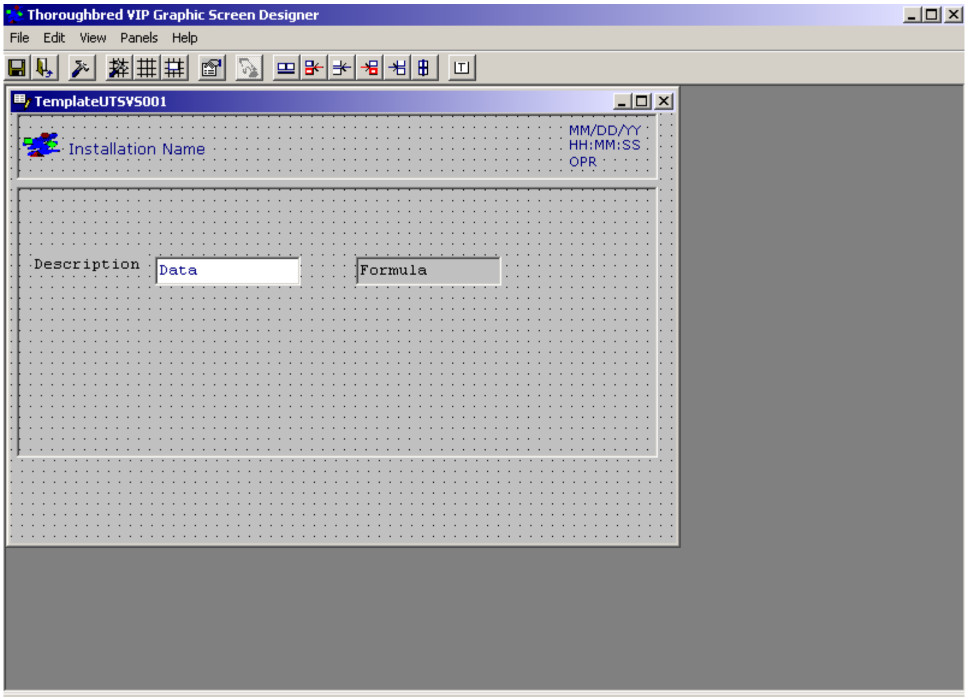
IMPORTANT: Do not delete UTSV001. See Create, Copy, Rename and Delete Templates section below.

For performance, templates are cached on the client. When cached on the client the current language code and time stamp are appended to the template name. For example the template UTSV001 on the host might be cached on the client as UTSV001-EN-0515031871001.

A template can contain the following objects:

- **Label control** - Used to apply font and color properties to all text on a screen.
- **Input control** - Used to apply font properties to all input fields on a screen.
- **Header panel** - Used to apply a header panel to a screen. Typically header panel contains the company logo, company name, date and time.
- **Data panel** - Used to apply a data panel to a screen. Typically a data panel contains all the screen input fields.
- **Footer panel** – Used to apply a third optional panel.
- **Image controls** – Used to add images that are common to all screens.
- **Button controls –** Used to add command buttons that are common to all screens.

The following image is the template UTSV01 loaded in the Graphic Screen Designer:



Applying Templates To Screens

Selecting a template to apply to a default graphical screen at runtime is done in the WorkStation Manager. For more information see the WorkStation Manager manual (Screen Template Options tab). Only templates that are cached on the client will be available for selection. To cache a template on the client or to update an existing template cached on the client, load the template in the Graphic Screen Designer then exit. It is not necessary to save the template back to the host, simply loading it in the Designer will update cache.

When a list of templates is displayed for selection, the extended template names are listed: template name, language code, and time stamp. For example: UTSV001-EN-0515031871001.

The template is only applied when a default screen is generated. If a default screen is cached on the client, the template will not be applied. To clear local cache make a slight modification to the corresponding character screen definition. This will update the last change date and time in the screen header, forcing a new default screen to be generated for all VIP clients.

NOTE: If you decide to change to a different template, all controls that were created by that template application will be destroyed prior to applying the new template. The screen text, input fields, and formulas are left in tact but will have the properties defined in the new template applied to them.

For information on applying templates to custom graphical screens see Graphic Screen Designer.

Create, Copy, Rename and Delete Templates

New templates are not created from scratch. To create a new template, on the host copy an existing template definition. The default template UTSV001 has been provided for this purpose. UTSV001 can be used to create (copy) custom template definitions. Follow standard Dictionary-IV copy procedures to copy a screen definition.

To rename a template definition, rename the template definition on the host using standard Dictionary-IV procedures to rename a screen definition.

To delete a template definition, delete the template on the host using standard Dictionary-IV procedures to delete a screen definition. In this case you will be given the option to delete both the character and graphical screen or just the graphical screen.

To access a new, copied, or renamed template for editing, load the template in the Graphic Screen Designer.

Backup/Restore Templates

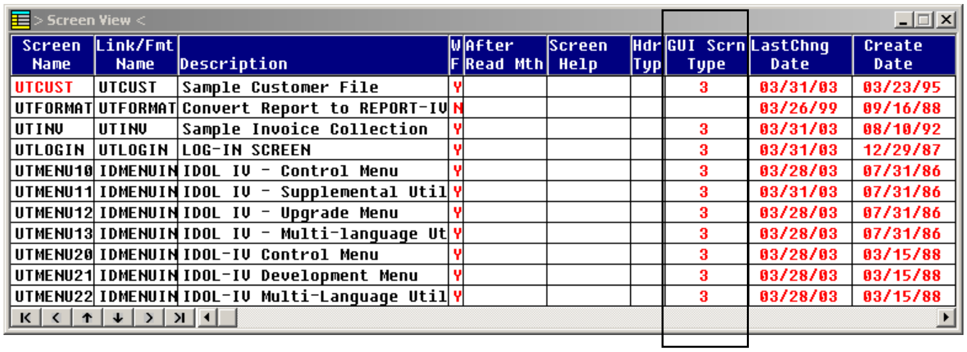
To backup or restore a template, select the corresponding character screen definition. For example to backup the template UTSV001, simply select the screen UTSV001 for backup. For more information, see backup/restore of graphical screens in this manual.

GRAPHIC SCREEN DESIGNER

IMPORTANT: The VIP GUI Screen Editor is not supported on Windows Vista, Windows Server 2008 and Windows 7.

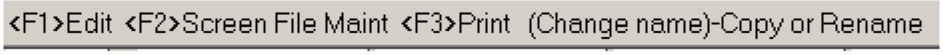
The VIP Graphic Screen Designer is used to create and edit both custom graphical screens and template definitions. It is used to enhance the visual presentation of a graphical screen. Changes that effect screen processing logic such as data element lists, data element types, and data element attributes cannot be made from within the screen designer. Any changes that impact the screen logic must be made to the format definition or to the character screen definition. Updates to formats and character screens are also made to corresponding graphical screen definitions.

The screen designer will be automatically executed on the client from the host through Dictionary-IV screen maintenance. With VIP enabled, select **Dictionary-IV** then **Screens**. The screen definition view will be displayed. The **GUI Scrn Type** column will indicate if a graphical screen exists. Custom graphical data entry screens are type 3. Templates are type 4. If a graphical screen does not exist, this column will be blank.



Screen Name	Link/Fmt Name	Description	W F	After Read Mth	Screen Help	Hdr Typ	GUI Scrn Type	LastChng Date	Create Date
UTCUST	UTCUST	Sample Customer File	Y				3	03/31/03	03/23/95
UTFORMAT	UTFORMAT	Convert Report to REPORT-IV	N					03/26/99	09/16/88
UTINV	UTINV	Sample Invoice Collection	Y				3	03/31/03	08/10/92
UTLOGIN	UTLOGIN	LOG-IN SCREEN	Y				3	03/31/03	12/29/87
UTMENU10	IDMENUIN	IDOL IV - Control Menu	Y				3	03/28/03	07/31/86
UTMENU11	IDMENUIN	IDOL IV - Supplemental Util	Y				3	03/31/03	07/31/86
UTMENU12	IDMENUIN	IDOL IV - Upgrade Menu	Y				3	03/28/03	07/31/86
UTMENU13	IDMENUIN	IDOL IV - Multi-language Ut	Y				3	03/28/03	07/31/86
UTMENU20	IDMENUIN	IDOL-IV Control Menu	Y				3	03/28/03	03/15/88
UTMENU21	IDMENUIN	IDOL-IV Development Menu	Y				3	03/28/03	03/15/88
UTMENU22	IDMENUIN	IDOL-IV Multi-Language Util	Y				3	03/28/03	03/15/88

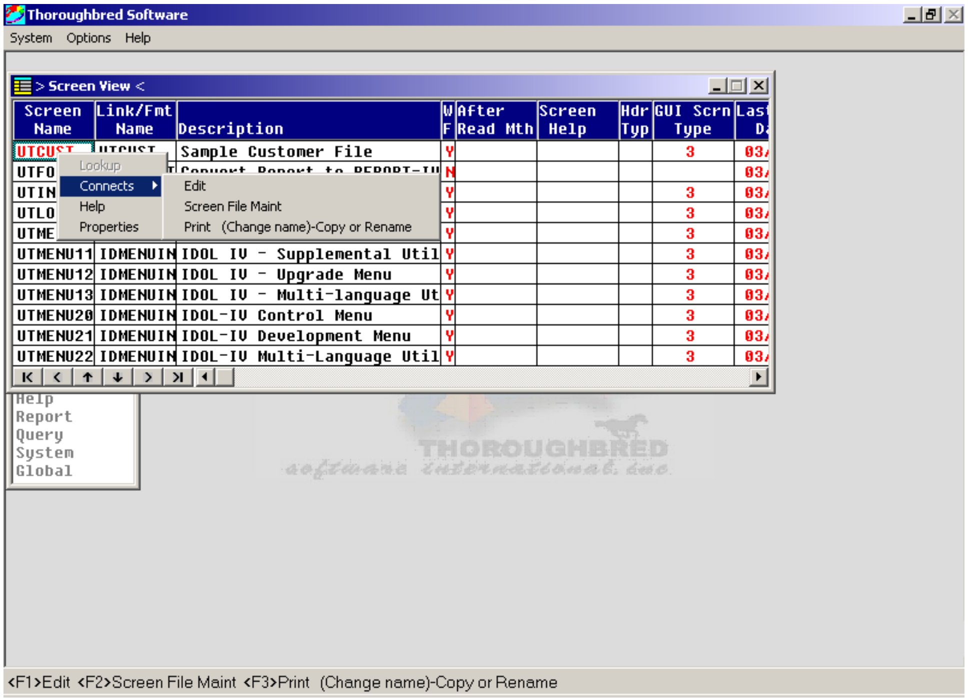
Highlight the screen or template file name (in the example above UTCUST). The options appear in the status bar.



<F1>Edit <F2>Screen File Maint <F3>Print (Change name)-Copy or Rename

Press **F1** to edit the screen.

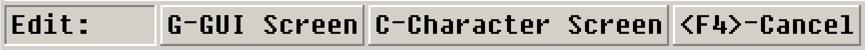
NOTE: VIP allows right mouse clicks to provide alternative ways to select options. A right mouse click on the screen file name provides the options seen below.



Screen Name	Link/Fmt Name	Description	WF	After Read Mth	Screen Help	Hdr Typ	GUI Scrn Type	Last D
UTCUST	UTCUST	Sample Customer File	Y				3	03/
UTFO	T	Convert Report to REPORT-IV	N					03/
UTIN			Y				3	03/
UTLO			Y				3	03/
UTME			Y				3	03/
UTMENU11	IDMENUIN	IDOL IV - Supplemental Util	Y				3	03/
UTMENU12	IDMENUIN	IDOL IV - Upgrade Menu	Y				3	03/
UTMENU13	IDMENUIN	IDOL IV - Multi-language Ut	Y				3	03/
UTMENU20	IDMENUIN	IDOL-IV Control Menu	Y				3	03/
UTMENU21	IDMENUIN	IDOL-IV Development Menu	Y				3	03/
UTMENU22	IDMENUIN	IDOL-IV Multi-Language Util	Y				3	03/

<F1>Edit <F2>Screen File Maint <F3>Print (Change name)-Copy or Rename

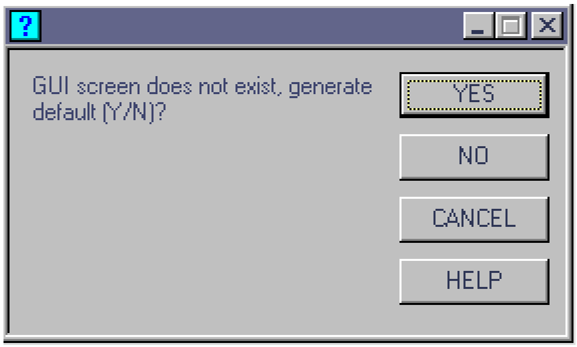
When you select **Edit** the following options appear in the status bar (or message dialog box).



Edit: G-GUI Screen C-Character Screen <F4>-Cancel

Select **G-GUI Screen**.

If the graphical screen definition does not exist, the following options appear in a message dialog box or the status bar.

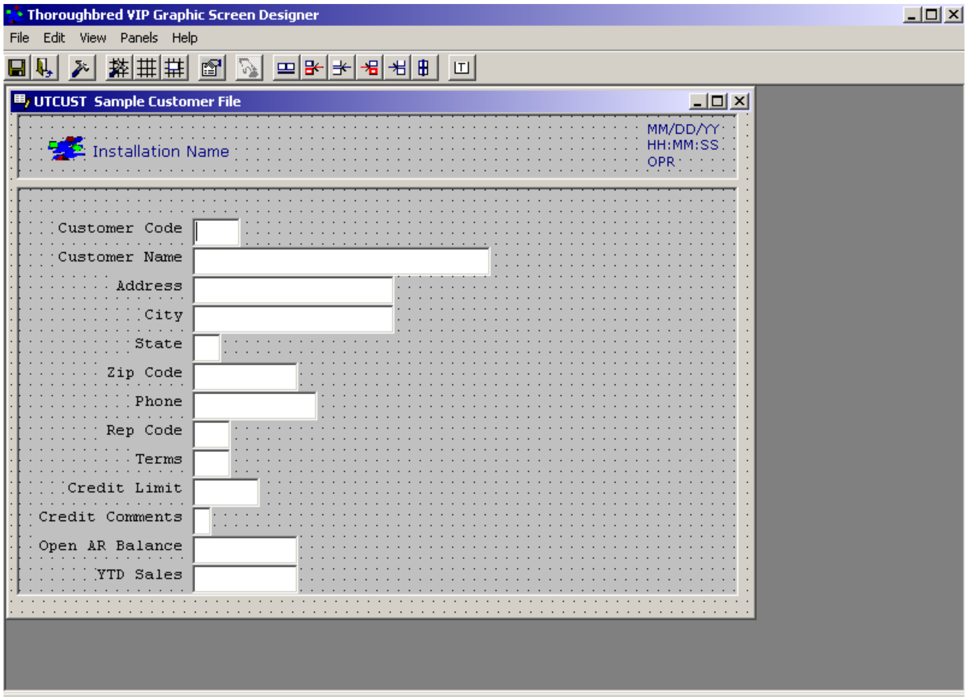


GUI screen does not exist, generate default (Y/N)?

Select **Yes** to generate a default graphical screen. If a default template option has been set in the WorkStation Manager, that template will be applied when the default screen is generated. See WorkStation Manager (Screen Template Options tab). Templates can also be applied from within the Graphic Screen Designer; it is not necessary to set the template option in the WorkStation Manager.

TIP: The generated default screen can be stylized through the use of templates and the screen filter file. Using these tools can significantly reduce the amount of editing from within the Graphic Screen Designer. Please refer to the Create Graphic Presentations (Screens, Templates, Screen Filters) sections in this manual.

The Graphic Screen Designer will be started and the graphical screen or template will be loaded.



The previous screen is based on the template installed with VIP for Dictionary-IV (UTSV001).

When a graphic screen or template is loaded in the Graphic Screen Designer, the host will remain in a "wait state." The prompt message "Host/Screen Editor Communication in progress…" will display while the host remains in the wait state. The communication link between the client and the host must remain open until the graphic screen or template is uploaded or unloaded from the editor. When a graphical screen is uploaded from the designer to the host, this prompt message is cleared. Only then is the host available to perform other Dictionary-IV functions. For more information see the Saving, Unloading Screens, Closing the Designer section in this manual.

NOTE: The process for editing screen templates is the same as editing custom graphical screens with some special considerations. These are noted where appropriate.

Menus and Toolbar Buttons

There are two buttons that do not appear in the menu option.

The **Shift Down**, **Jump to Font** button enables mouse shortcuts for control objects. When this feature is enabled, clicking the mouse (left or right mouse button) on an object with text properties displays the font properties dialog box. Clicking on an object with no text properties (e.g.: form panels) brings up the color properties dialog box. Press the button again to disable these shortcuts.

Use this button to apply a template to the current screen. To select the template definition, see the Options (Templates) section below. A template should not be applied to a template definition, only to a custom graphical screen.

Below are the menus options and tool bar buttons (represented graphically) available in the Graphic Screen Designer.

Files
Perform one of the following:

Save Screen

Save Template

From the File menu, only one of the save options will be enabled depending upon the current screen type.

For more information on saving graphical screens see Saving, Unloading Screens, Closing the Designer in this manual.

Exit (disconnect from host and exit)

Edit
Options allow you to set various preferences and options for the designer. The Options dialog contains three tabs (General, Templates, and Command Button). For more information see the Options section in this manual.

View
Perform one of the following:

Open/Close Object Tool box

Open/Close Grid Tool box

Open/Close Object Property List

For more information see Object Toolbox, Grid Tool box, and Object Property List sections in this manual.

Panel

Perform one of the following:

Add or Delete a Header Panel

When a Header Panel is added to a screen, it is placed at the top of the screen. To resize or move the Header Panel use normal Windows resizing techniques (drag edges left/right or up/down).

If a Data Panel exists, by default the width of the Header Panel will be sized to the width of the Data Panel. To enable/disable this option please see Options, Templates. A similar option is available at runtime when a template is applied to a default screen. See the WorkStation Manager manual (Screen Template Options tab).

When a panel is deleted from a screen, all controls except image controls will be moved from the panel to the screen form. Image controls must be manually moved off the panel before deleting it. Any image controls left on the panel will be deleted.

Add or Delete a Data Panel

When a Data Panel is added to a screen, it is placed at the top of the screen. To resize or move the Header Panel use normal Windows resizing techniques (drag edges left/right or up/down).

When a panel is deleted from a screen, all controls except image controls will be moved from the panel to the screen form. Image controls must be manually moved off the panel before deleting it. Any image controls left on the panel will be deleted.

Do not delete the Data Panel from a template definition. All templates must contain a Data Panel.

Add or Delete a Footer Panel

When a Footer Panel is added to a screen it is placed at the bottom of the screen or below a Data Panel. To resize or move the Header Panel use normal Windows resizing techniques (drag edges left/right or up/down).

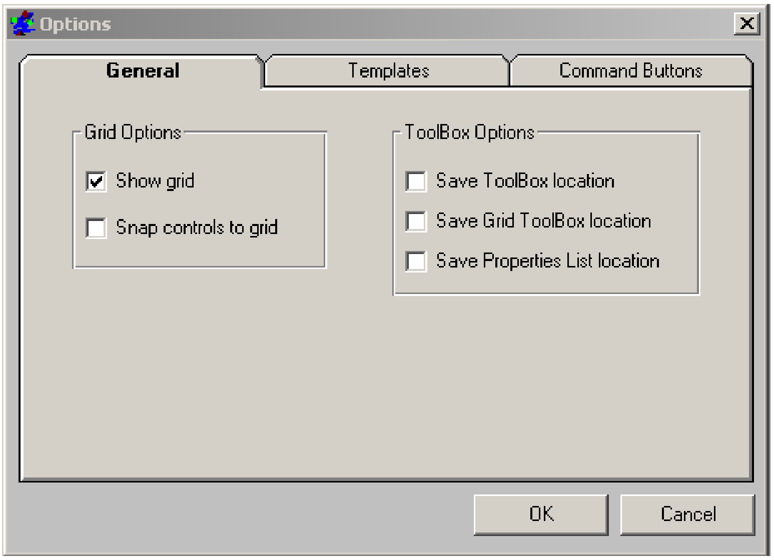
If a Data Panel exists, by default the width of the Footer Panel will be sized to the width of the Data Panel. To enable/disable this option please see Options, Templates. A similar option is available at run time when a template is applied to a default screen. See the WorkStation Manager manual (Screen Template Options tab).

When a panel is deleted from a screen, all controls except image controls will be moved from the panel to the screen form. Image controls must be manually moved off the panel before deleting it. Any image controls left on the panel will be deleted.

Help
Use this menu to view Online, Technical Support, and About help.

Options

The Options selection opens the following window with three tabs: General, Template, and Command Buttons.



Grid Options

☑ Show grid

□ Snap controls to grid

ToolBox Options

□ Save ToolBox location

□ Save Grid ToolBox location

□ Save Properties List location

General Tab

Show grid
Displays a grid on the screen.

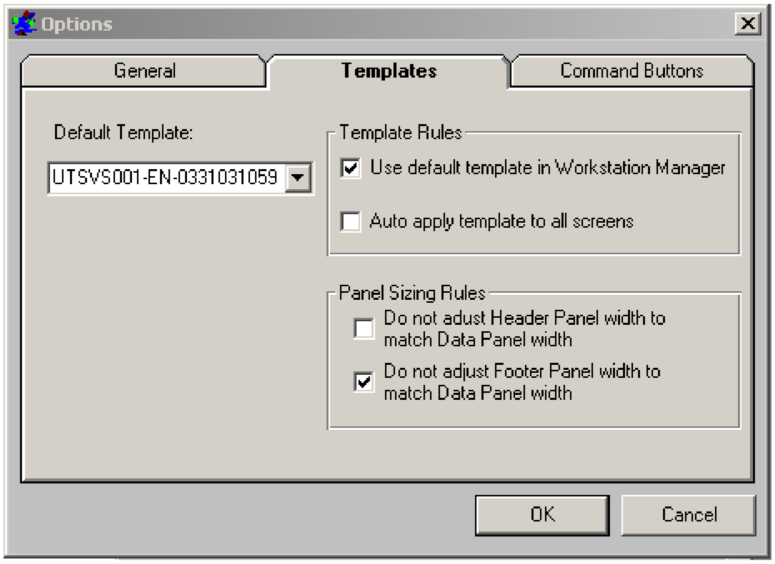
Snap controls to grid
Enable and disable the **Snap To Grid**.

Save Toolbox location
The Object Toolbox will open at the location where it was last closed.

Save Grid Toolbox location
The Grid Toolbox will open at the location where it was last closed.

Save Properties List location
The Properties List will open at the location where it was last closed.

Templates Tab



Default Template: UTSVS001-EN-0331031059

Template Rules

☑ Use default template in Workstation Manager

□ Auto apply template to all screens

Panel Sizing Rules

□ Do not adust Header Panel width to match Data Panel width

☑ Do not adjust Footer Panel width to match Data Panel width

Default Template
This drop down list box allows you to select a template that can be applied to screen definitions in the Graphic Screen Designer. This enables you to specify a different template from the one specified in the WorkStation Manager. The installed default is shown above. Only template definitions cached on the client will be available for selection. To cache a template definition, load that template in the screen designer from Dictionary-IV screen maintenance.

Use default template in the WorkStation Manager
When selected, the editor will use the default template that has been specified in the WorkStation Manager.

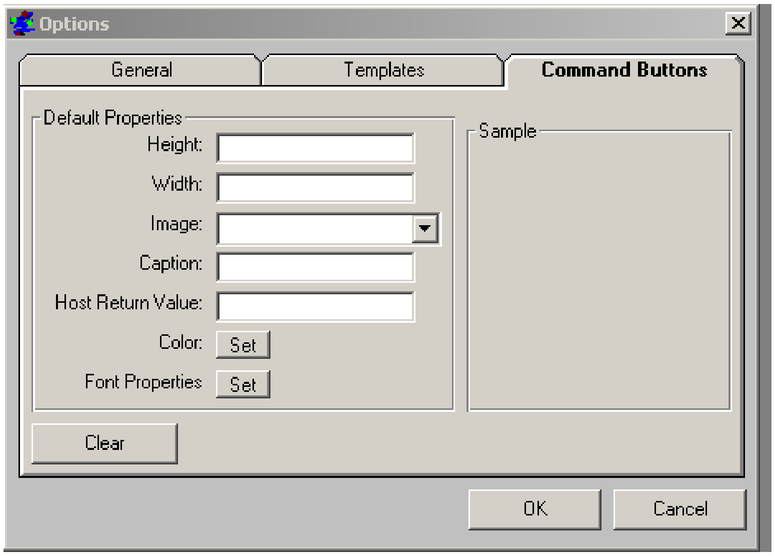
Auto apply template to all screens
When enabled, the default template will be applied to all screens when they are loaded in the editor. If this is not enabled, a template will only be applied when the apply template button is pressed.

NOTE: In addition to the two template options described above, when a default screen generated for the designer and the WorkStation Manager option to automatically apply a template is set, that template will be applied to the default screen. See WorkStation manager manual (Screen Template Options tab).

Do not adjust Header Panel width to match Data Panel width
When a Header Panel is added, it will not be auto sized to the width of the Data Panel.

Do not adjust Footer Panel width to match Data Panel width
When a Footer Panel is added, it will not be auto sized to the width of the Data Panel.

Command Buttons Tab



The dialog allows you define the default properties to be used when creating new command button controls.

Command Button Default Properties

It is possible to set default properties to be applied to a Command Button when it is first created. If an entry does not have a value, it is ignored. You can specify any or none of the default properties (Height, Width, Image, Caption, Host Return Value, Color, and Font).

Example Command Button Entries

`cmdbuttbackcolor=&H00C0C0C0`
This entry allows you to specify the background color of the Command Button. This must be hex notation. The default background color is the system default color in the user’s preferences specified in the system desktop appearance setting.

`cmdbuttcaption=my button`
This entry allows you to specify the button caption.

`cmdbuttheight=750`
This entry allows you to specify the button height.

`cmdbuttwidth=750`
This entry allows you to specify the button width.

`cmdbuttimage=tsi.ico`
This entry allows you to specify the image (picture property). The image file must be located in *vip-data-path*/Tsi/Image folder.

`cmdbuttfontname=Arial`
This entry allows you to specify the font name for the caption.

`cmdbuttfontsize=8`
This entry allows you to specify the font size for the caption.

`cmdbuttfontbold=T`
This entry allows you to specify the font bold property, **T** (true) for bold and **F** (false) for regular.

`cmdbuttfontitalic=T`
This entry allows you to specify the font italic property, **T** (true) for italic and **F** (false) for regular.

`cmdbutthostrv={F9}`
This entry allows you to specify the Host Return Values. These must be entered exactly as you would from within the Graphic Screen Designer.

The following table contains valid values for the Background property:

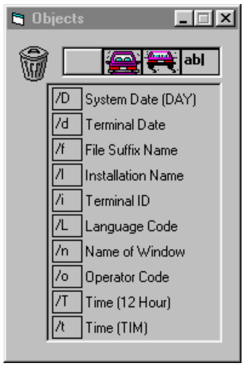
White	**Light Gray**	**Gray**	**Dark Gray**	**Darkest Gray**	**Black**
FFFFFF	E0E0E0	C0C0C0	808080	404040	000000
Lightest Red	**Light Red**	**Red**	**Dark Red**	**Darkest Red**	
C0C0FF	8080FF	0000FF	0000C0	000080	
Lightest Yellow	**Light Yellow**	**Yellow**	**Dark Yellow**	**Darkest Yellow**	
C0FFFF	80FFFF	40FFFF	40C0C0	408080	
Lightest Green	**Light Green**	**Green**	**Dark Green**	**Darkest Green**	
C0FFC0	80FF80	40FF00	40C000	408000	
Lightest Cyan	**Light Cyan**	**Cyan**	**Dark Cyan**	**Darkest Cyan**	
FFFFC0	FFFF80	FFFF00	C0C000	808000	
Lightest Blue	**Light Blue**	**Blue**	**Dark Blue**	**Darkest Blue**	
FFC0C0	FF8080	FF0000	C00000	800000	

Toolboxes

Object Toolbox

The Object Toolbox is used to add and delete controls on the screen from.

Select the **Object Toolbox** from the View option on the menu bar, or select the **Object Toolbox** button from the toolbar.



/D	System Date (DAY)
/d	Terminal Date
/f	File Suffix Name
/I	Installation Name
/i	Terminal ID
/L	Language Code
/n	Name of Window
/o	Operator Code
/T	Time (12 Hour)
/t	Time (TIM)

From the Object Toolbox you can drag and drop buttons, images, text, or any of the listed Dictionary-IV special characters. Once added, you can use Object Properties to set the attributes of the objects (see Object properties later in this manual).

Delete Objects

Trashcan
One of the first items you'll see is the trashcan. Drag and drop objects you want to delete into the trashcan. When editing templates you cannot delete the "Description" label, the "Data" control, or the "Formula" control. These are required on all templates.

Add Buttons, Images, and Text

Command Button
Drag and drop this button to any position on the screen. From the Object Properties List you can set the properties for this command button.

Fixed Image
Drag and drop this button to any position on the screen. From the Object Properties List select the image attribute. The image screen appears. You can select an icon (.ico) or a bitmap (.bmp or .dib) image. You can resize and crop the image container control. Resize the image by grabbing a side and stretching it. This will not resize the image, only the image container.

Resizable Image
Drag and drop this button to any position on the screen. From the Object Properties List select the image attribute. The image screen appears. You can select an icon (.ico) or a bitmap (.bmp or .dib) image. After you select an image you can resize it by grabbing a corner and stretching. This maintains the aspect ratio (resizes proportionally).

abl

Text
Drag and drop this button to any position on the screen to add text. Use the Object Properties list to define any of the properties (such as color and font) associated with text objects.

Special Screen Codes

The Object Toolbox allows you to select the Dictionary-IV special screen codes and place them on your graphic screen:

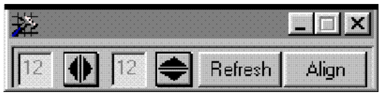
/D	System Date (Thoroughbred Basic directive DAY)
/d	Terminal Date
/f	File Suffix Name
/I	Installation Name
/I	Terminal ID
/L	Language Code
/n	Name of Window
/o	Operator Code
/T	Time (12 Hour)
/t	Time (Thoroughbred Basic directive TIM)

Select a special screen code. Drag it to the appropriate position on the screen and drop it. Select the **Properties List** to define the attributes of the special screen code.

The current host values for special screen codes (i.e. /D, /I etc.) are not displayed while editing a screen. Descriptive text is used for the special screen codes. At runtime, the descriptive text is replaced with the current host value. This text is maintained in the workstation message file. For more information on editing the message file see the Multi-Language for VIP manual.

Grid Tool

Select the **Grid Tool** option from the View menu. The system displays the following:



From this tool you can select the horizontal and vertical size of the grid (measured in pixels). You can also **Refresh** the grid and **Align** objects to the grid.

NOTE: Using the Grid Tool with the option Snap to Grid enabled can cause controls to "jump" when dropped on the form. Disabling this option and using the control alignment options described below provides more precision when moving controls on the form.

MenuBar Buttons

You can select multiple controls by holding the **Control** key while selecting the controls.

Y-axis align: Aligns selected controls along the Y axis, position all controls based on the bottom most control

X-axis align: Left align selected controls along the X axis, position all controls based on the position of the first selected control.

X-axis align: Left align selected controls along the X axis, position all controls based on the position of the left most control.

X-axis align: Right align selected controls along the X axis, position all controls based on the position of the first selected control.

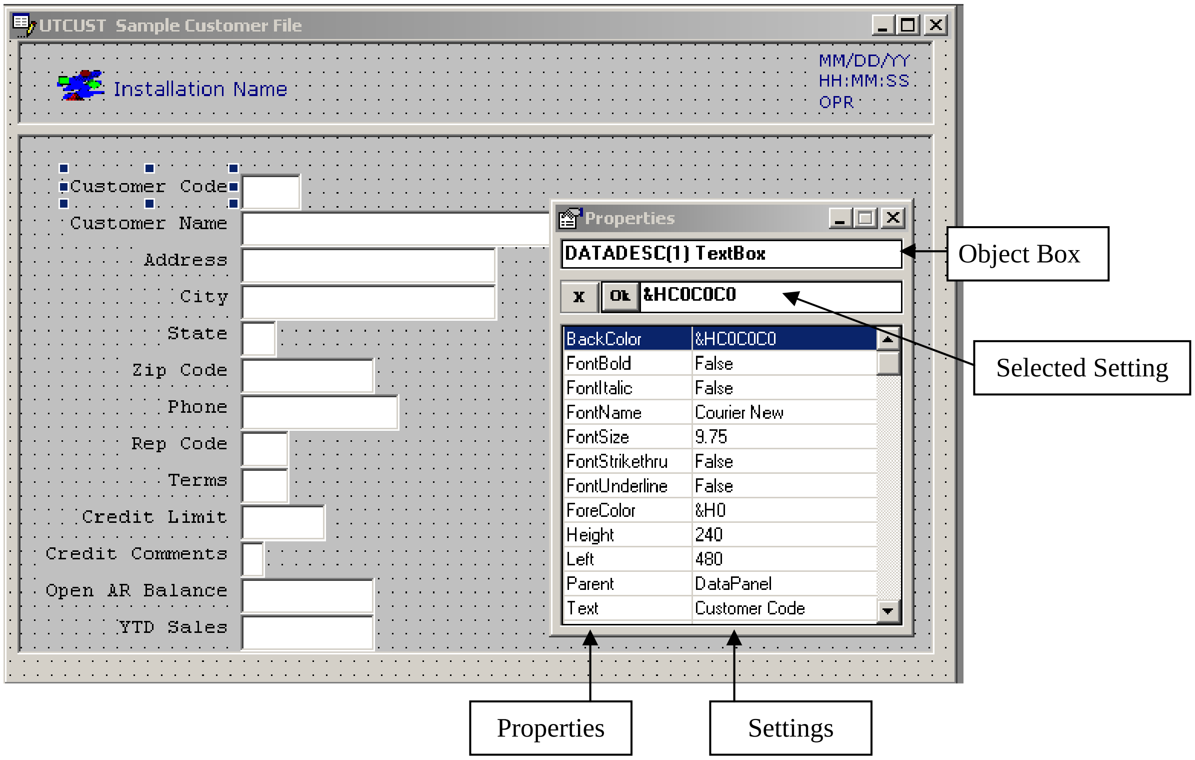
X-axis align: Right align selected controls along the X axis, position all controls based on the position of the right most control.

X-axis align: Center align selected controls along the X axis.

NOTE: These allow you to properly align the controls on the screen with the **Snap To Grid** disabled.

Object Property List

The Object Properties list provides you with a method of viewing and setting optional properties for objects on the graphic screen. Select the **Object Property List** option from the **View** menu or click on the Object Property List button on the menu bar. Select the text "Customer Code" and the property listing for the textbox control (label) appears. See below.



DATADESC(1) TextBox

x | Ok | &HC0C0C0

BackColor	&HC0C0C0
FontBold	False
FontItalic	False
FontName	Courier New
FontSize	9.75
FontStrikethru	False
FontUnderline	False
ForeColor	&H0
Height	240
Left	480
Parent	DataPanel
Text	Customer Code

The attributes are a function of the object. The content of the box varies with the type of object. The Properties List is designed in a Visual Basic style.

Object Box

Contains the name of the object for which you can set properties. In the example above DATADESC(1) Text Box. The object name cannot be changed.

Property List

Displays all of the properties for the screen object. The right column displays the current settings for each property. In the example above it is the BackColor and the color code &HCOCOCO, which is highlighted.

NOTE: Not all properties can be changed. Properties exposed for editing is a function of the object type.

Setting Box

Lets you edit the settings for the property you select in the property list. To save a change made to a property in the setting box, click the **Ok** button. To cancel a change made to a property click the **X** button or click on another property in the property list.

- Color Settings - In the example above double-click on &HCOCOCO in the Setting Box. The Windows color dialog displays where you can select a different color.

- Font Settings - If you move the selection in the properties list to FontName Courier New and double-click on Courier New, the Windows font dialog displays, where you can change the font and its properties.

- True/False Settings – If you move the selection in the properties list to FontBold False, you can use the arrow to select **True**.

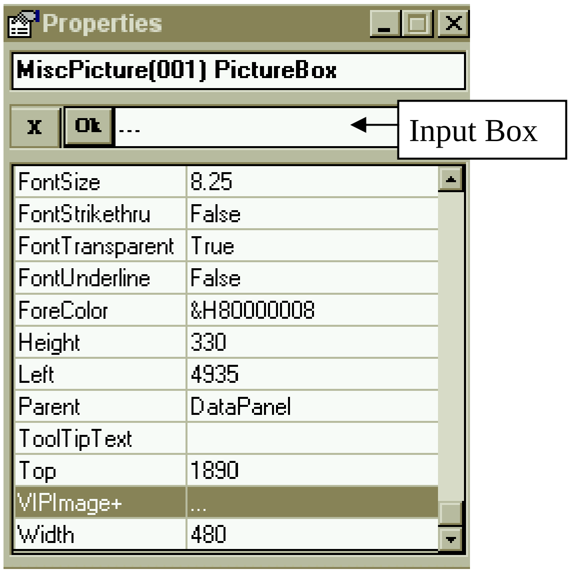
- Input Settings – If you highlight Text Customer Code, you can type over "Customer Code" to change the text in the window. The same holds true for Height, Width, Left, etc.

- Image/Button Settings – Use the same method to select properties for images. For information about adding actions to images or buttons, see the following section.

Adding Actions to Images or Command Buttons

Select the image or command button from the Object Toolbox.

Drag and drop it on the graphic screen or template. Open the Properties List and select the image and the following window opens.

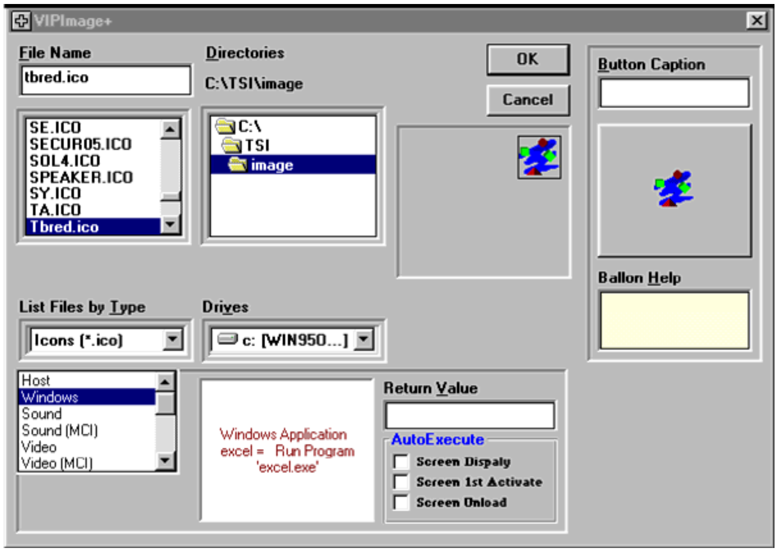


MiscPicture(001) PictureBox

...

FontSize	8.25
FontStrikethru	False
FontTransparent	True
FontUnderline	False
ForeColor	&H80000008
Height	330
Left	4935
Parent	DataPanel
ToolTipText	
Top	1890
VIPImage+	...
Width	480

Double-click on the input box to display VIPImage+. Images you want to use should be stored in the *vip-data-path* \image directory. In the example below, the Tbred.ico file was selected; however, many image file types are supported.



File Name: tbred.ico

Directories: C:\TSI\image

SE.ICO
SECUR05.ICO
SOL4.ICO
SPEAKER.ICO
SY.ICO
TA.ICO
Tbred.ico

C:\
TSI
image

List Files by Type: Icons (*.ico)

Drives: c: [WIN950...]

Host
Windows
Sound
Sound (MCI)
Video
Video (MCI)

Windows Application
excel = Run Program
'excel.exe'

File Name

Select the graphic from the *vip-data-path* \image folder.

List Files by Type

Select the graphic file type you are looking for (bmp, dib, ico, etc.).

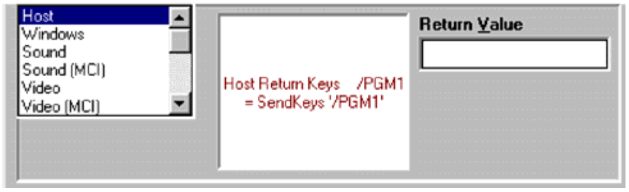
Button Caption

Allows you to display text with your image or button.

Balloon Help

Allows you to display help when the cursor passes over the image or button.

At the bottom of the window you will find the following.



Host
Windows
Sound
Sound (MCI)
Video
Video (MCI)

Host Return Keys /PGM1
= SendKeys '/PGM1'

From here you can set the action for the image or command button. For more information see Toolbar Maintenance in the WorkStation Manager manual. You can set the action to take place when the Screen Displays, Screen first Activates, or when the Screen Unloads.

When a graphical screen definition is loaded and it contains an image control that references a missing image file, the image control is not visible. The image control itself has no visible properties; only the picture contained by the control is rendered. Consequently an empty image control is enabled but it is not visible.

To help locate and identify an image control with an invalid image file name, a warning message will be displayed. This message displays the name of the missing image file and gives you two options:

OK to set image name to missingimagefile.bmp. Selecting this option will update the image control setting the image file to missingimagefile.bmp. This image is similar to the image used when an image control is first added to a screen. By replacing the invalid image file with a valid one, the image control is visible and can be easily corrected. If the screen definition is saved, it will be saved with missingimagefile.bmp

Cancel to load the empty image control. Selecting this option will load the image control but it will not be visible. missingimagefile.bmp is displayed so that the control is visible but is not saved with the definition.

OK
Select the OK button on the VIP Image+ screen to save your edits.

Cancel
Select the Cancel button on the VIP Image+ screen to exit the widow without saving your edits.

Saving, Unloading Screens, Closing the Designer

To save a custom graphical screen or template back to the host, select one of the following save options:

- From File select **Save Screen** or **Save Template**
- From the menu bar click the **Save** button
- From the menu bar click the **Exit** button. This option will also close the Graphic Screen Designer.
- On the graphical screen from, click the close button (**X**) in the upper right corner of the screen form.
- Click the close button (**X**) in the upper right corner of the Graphic Screen Designer window. This option will also close the Graphic Screen Designer.

The following message dialog will display:

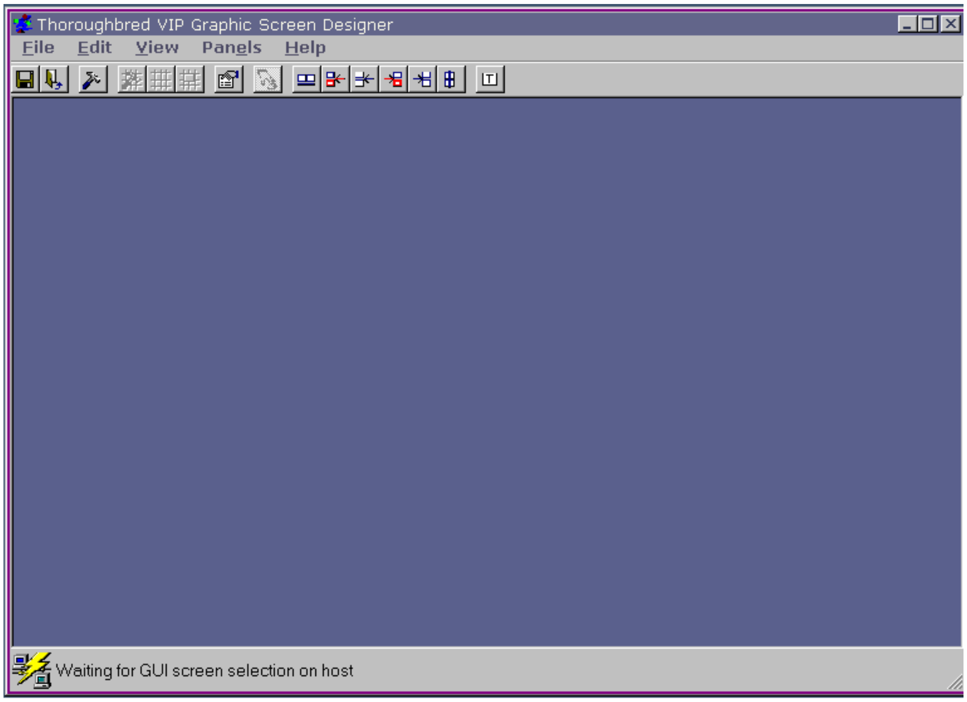
```
Save Changes to screen screen-name?
```

Select **Yes** to save the changes and upload the custom graphical screen or template to the host.

Select **No** to unload the graphical screen or template without saving the changes.

Select **Cancel** to continue editing the graphical screen or template.

If one of the options to close the screen designer has not been selected the focus will be set to the main VIP GUI Server window and the designer will display the following message in the screen designer status bar:



Waiting for GUI screen selection on host

While this status message is displayed, another graphical screen or template can be selected for editing on the host using Dictionary-IV screen maintenance. This eliminates the need for the Graphic Screen Designer to be loaded each time a graphical screen or template is selected for editing.

The designer can be closed at any time by clicking on the close button (**X**) in the upper right corner of the main Graphic Screen Designer window.

SCREEN FILTERS

Screen filters allow VIP to modify the text portion of a screen when generating default screens. Using screen filters will result in more presentable default and custom graphical screens.

The type of editing to be done to the screen text is controlled by a screen filter file. Screen filters are optional. If a screen filter file does not exist, no screen text editing is performed. Screen filters are global. That is, they apply to all screens whenever a default screen is generated.

The screen filter allows you to remove or replace characters originating in the character screen definition prior to generating a default graphical screen. This can be used to remove the "...:" typically found in character screen designs. The screen filter can also be used to clear entire rows from the character screen when generating a default graphical screen. This is useful if in your graphical screen (through the use of a template) you want the screen codes such as the installation name, date, operator code, etc. to appear in a header panel and do not want them in the body of the screen. The screen filter can clear these rows of text before generating the default graphical screen.

Building a Screen Filter File

A default screen filter is distributed with VIP and placed in *vip-data-path* **\Tsi\Scripts\Scrfilter.txt**. Use Notepad.exe to edit the default screen filter file or create a new one. The screen filter file must have a .txt file extension.

The screen filter file must conform to the rules for a standard INI file. Comment lines begin with " ; " or " # " and may appear anywhere in the file. The first non-comment line must be:

```
[LABELEDIT]
```

Specifying a delimiter

A delimiter is a single character that is used to separate label text. In almost all cases this should be a space. To specify the delimiter to be used in a screen filter file add this line:

```
Delimiter='x'
```

where **x** is usually a single space. The single quotes (') are required.

Replacing characters in a text line

In order to replace characters in a text line you must have previously specified a delimiter. The character defined as a delimiter is used to replace specified characters. To indicate what characters are to be replaced, add this line:

```
Replace='abc...'
```

where **abc..** are any number of characters that are to be replaced in the text line. The single quotes (') are not part of the string and cannot be included in the string. VIP will search for each character in the string and replace it with the delimiter character if the character is found.

Clearing a line of text

The screen filter file can designate lines to be blanked out or cleared. The screen filter file allows for two clear line commands:

```
ClearLinStr1=string1

ClearLinStr2=string2

ClearLinStr1=^string1

ClearLinStr1=$string1

ClearLinStr1=~string1
```

where: **string1** and **string2** can contain any characters including embedded, leading, or trailing spaces, but they cannot contain the single quotes ('), which is used to delimit the string. Leading and trailing spaces are significant. The line will be removed if it matches exactly the specified string. The ^ character indicates the line is to be removed if it begins with 'string1'. The $ character indicates the line is to be removed if it ends with **string1**. The ~ character indicates the line is to be removed if it contains **string1** anywhere in the line.

Identifying labels

Normally VIP will delimit labels in a text line by a single space. For example, the line **Customer Name** will contain two labels **Customer** and **Name**. A filter file can change the way VIP parses labels with this line:

```
Rule1=Y
```

If this line is included in the screen filter file, a single space will be considered part of the label text. For example, **Customer Name** will result in one label control being created because of the single space between **Customer** and **Name**. If two or more spaces exist between **Customer** and **Name**, two label controls will be created.

How VIP Processes a Screen Filter File

VIP processes the **ClearLinStr** commands before any characters are replaced in the text string. If the Delimiter command is missing the Replace command has no effect.

Integration in the Tbred.ini file

If a screen filter file is to be used it must be specified in the Tbred.ini file. Add the following line to the [SCREENS] section of the Tbred.ini file:

```
Scrfilter=anyname
```

where *anyname* is the name of the screen filter file. This file can have any name and extension, but it must exist in the directory *vip-data-path*\Tsi\Scripts.

> **NOTE:** Even though the screen filter file can have any name and extension, it **must** be a plain text file, or a file readable as plain text.

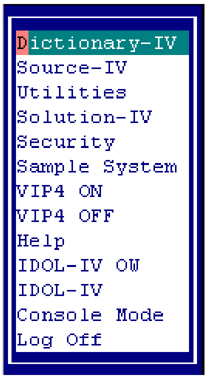
Sample Screen Filter File

```
[LABELEDIT]

#The delimiter is a space.

Delimiter=' '

#All space and ":" characters found in screen text lines will be
#replaced with a space.

Replace='.:'

#If a line contains "Company :"(Company followed by a space) or
#"Operator :" (Operator followed by a space) anywhere in the line, the
#entire line will be set to all spaces.

ClearLinStr1=~Company :

ClearLinStr2=~Operator :

#Text strings separated by a single space will be considered one label.

Rule1=Y
```

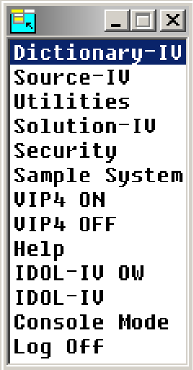
ENABLE THE GRAPHIC PRESENTATION

VIP can be enabled in one of several ways.

- To enable VIP from the main OPENworkshop menu:



Select **VIP ON** to enable VIP. It will remain enabled until the option **VIP OFF** is selected.



The VIP ON and VIP OFF options that are available from the main OPENworkshop menu (OOM0) cannot be added to another OPENworkshop menu. These selections are processed by the OPENworkshop start up program **OO**.

- To enable VIP at any point after an OPENworkshop or Dictionary-IV login call VIPCNT with MSG$[1]="GON". For example:

```
Dim MSG$[1];
MSG$[1]=”GON”;
CALL “VIPCNT”,MSG$[ALL]
```

- To enable VIP prior to the OPENworkshop and Dictionary-IV login modify the IPLINPUT file to run a custom OPENworkshop or Dictionary-IV start up program that does the following:

  ```
  DIM ]7$(400);                     ! Dim Dictionary-IV/OWS system variable
  CALL "IDPUB4",]6$,]7$,]8$,]9$;  ! Call Dictionary-IV system
  CALL "8ZPSYS",]SYSV$;           !    startup publics.
  ]7$(77,3)="  ",                 ! Clear or set operator code.
  ]7$(228,1)=" ";                 ! Clear or set auto login flag.
  A$=CGV("VIPSTART","ID");        ! Set VIP start up global to run ID
  A$=CGV("VIPSTART","OO");        !   or OO with VIP.
  RUN “VIPSTART”                  ! Run VIP start up.
  ```

  The program VIPSTART will enable VIP for Dictionary-IV and then run the program specified in the global variable VIPSTART. If the global variable is not set, the program VIPSTART will run **OO** (OPENworkshop) by default.

- To execute a host application menu selection from the Desktop, Start Button or from the WorkStation Manager, see Desktop Application Menus later in this manual.

DESKTOP APPLICATION MENUS

Dictionary-IV Menu selections can now execute from either the WorkStation Manager Applications option, the Windows Start button, the Desktop, or from within the VIP GUI Server Menu Bar.

Introduction to Desktop Application Menus

Selecting a Menu option from the Windows Start button executes Thoroughbred Basic with the values of the menu name and menu selection to be executed. The menu selection values are then passed to OOWEB for processing by the host.

If you execute menu selections from the VIP GUI Server Menu box, exiting the Menu selection goes back to the **OO** or **ID** menu program. The Thoroughbred Basic session remains active while at an input field. If you are running graphically, the GUI Server window remains visible and application menu selections can be made without starting another Thoroughbred Basic session. To exit the session press **F4**.

If you execute menu selections using any other method, exiting the Menu selection ends the Thoroughbred Basic session.

When executing your menu selections from the workstation, the Dictionary-IV/OPENworkshop startup program must be **OOWEB**. This must be added to the **iplooweb.txt** file.

```
IPL 1,2,T0,OOWEB
```

For more information see AutoLogon.

Creating Desktop Application Menus

To create your application menu selections for the workstation, you must convert your Dictionary-IV screen menu definitions to pop-up menu definitions. Once this is done a data file (**IDDBDWB**), containing all menu selection information and structure, must be generated. Utilities have been provided for this purpose.

1. Start Thoroughbred Basic using a standard **IPLINPUT** file. **RUN"OO**" and select **Utilities**.

2. Select **Build Menu Headers**. Select the **All** or **Select Libraries** option. This builds pop-up menu headers for your Dictionary-IV Menu libraries. If you choose **Select Libraries**, the system prompts for the library names. You may list multiple libraries separated by commas (L1, L2, L3, Ln).

3. Select **Build List Box Menus** and select your Dictionary-IV Menu Libraries. Enter a list of 2-character menu libraries or a list of menu names delimited by a comma (,). This generates pop-up menu definitions based on the selected Dictionary-IV menu libraries or menu definitions.

   The system creates Help modules matching the Screen menu name and overwrites Help modules with like names. The system creates multiple help names for Screen menus containing sub-headings. These are Menu entries with a blank **Selection** value and have **SH:** as the first 3 characters of the library/screen menu specification. Following is the window that appears:

```
Enter [SH:](LL,LL) or [SH:](LLM1,LLM2,LLMn)

___________________________________________________________________

Enter list of two character library names or menu names (LLMMMMMM)
delineated by commas. Combinations of library names and full menu names
will be allowed.

Note: Help modules matching the Screen Menu name will be created and will
overwrite help modules with like names. Multiple help names will be
created for Screen Menus containing Sub-headings (menu entries with a
blank selection value and SH: was first three characters of library/screen
menu specification). A menu help module will be created containing the
Sub-headings and one containing the Selections subordinate to each sub-
heading. The manes of the help modules will be as follows:


  LL    LLMM, LLMM0, LLMM1, LLMMn

  LLM1   LLM1, LLM10, LLM11, LLMMMn
```

4. Select **Build Start Menus**. This option generates the **IDDBDWB** file, which contains your menu structure and all the necessary information to process your application selections.

   The system displays a view of all libraries. Scroll to the desired library name and press **Enter** to select it.

   The system displays a view of all pop-up menu definitions. Scroll to the desired menu definition and press **Enter** to select it.

   At this point you have two options:

   - Press **Enter** to generate the menu selection data and preserve your menu tree structure. (This generates a context or window menu for each menu referenced by the root menu. This emulates the same menu structure or menu paths as in Dictionary-IV.)

   - Press **F2** to generate one single context or window menu. (This generates a single context menu containing all menu selections for each menu.)

Select **Set Start Menu**. The system displays a view of menus that have been processed by the **Build List Box Menus** option and the **Build Start Menus** option.

If you are using Dictionary-IV menus (**ID**), you will need to modify the text found in the **Action** column. Move the cursor to the **Act** column and press **Enter**. Add the following to the end of the first line of text.

```
EXEC=ID
```

The first line of text in the **Action** column for a Dictionary-IV menu should be similar to the following:

```
CONNECT MENU menu-name EXEC=ID
```

Return to the **Menu Name** column. Press **F1** to add the menu to both the WorkStation Manager Applications and to the Windows Start button. Press **F2** to add the menu to the GUI Server Menu Bar.

NOTE: you must first select **F1**, the **F2** option is only valid after having selected **F1**.

You can now press **F4** to exit Basic.

Create an **IPLINPUT** file to be used with the Start button, WorkStation Manager, or Desktop shortcuts. In the Thoroughbred home directory (i.e**. \TBC**), copy **IPLINPUT** to **IPLWEB**. Edit **IPLWEB** and make the following changes.

Change the startup program from **ID** to **OOWEB**. See the example below.

```
IPL 1,2,T0,OOWEB
```

If you are running on a Remote Host

Define an Au7toLogon entry on the WorkStation Manager Options tab.

If you are running on a Local Host

Be sure you have properly modified the **Tbred.ini** file for the **startbasic** entry as described in Installation.

WorkStation Manager Options

NOTE: Before executing your application menus, verify that the WorkStation Manager options are setup properly. To get started, check the Host Type (**Remote** or **Workstation**) and the Presentation Mode (**Character** or **VIP**).

VIP TOOLBAR

VIPTBAR is a Method provided by Thoroughbred Software that allows you to control toolbar presentation and behavior from within your application software. VIPTBAR uses the toolbar definitions maintained by the WorkStation Manager please see the WorkStation Manager manual (Toolbar Maintenance option).

In VIP you can create and edit toolbars using Toolbar Maintenance. When you create a toolbar VIP creates two files:

- A toolbar definition file with a def extension.
- An image file with a bmp extension.

VIP then stores them in the *vip-data-path* \ToolBar\WsnMgr folder.

All images used by the toolbars must be stored in the *vip-data-path*\ToolBar folder.

VIP shares the same toolbar definitions as the WorkStation Manager toolbar but displays them inside the main VIP GUI Server window.

VIP also supports a dynamic toolbar stack. Host application code can create and manage a stack of multiple toolbar definitions and dynamically control which toolbar definition is available to the user. For example, it is possible to have one toolbar displayed when doing screen processing and another toolbar displayed with doing view processing.

The VIPTBAR Method performs one of five functions:

- Define a toolbar. This function exposes an existing toolbar definition to VIP for subsequent processing. Defining a toolbar adds a toolbar definition to the toolbar stack, it does not activate a toolbar.
- Modify a toolbar. Modify some behavior of the toolbar such as location, docking options, floating options, ToolTip options, and hide and pop options.
- Activate a toolbar and change its location.
- Deactivate a toolbar.
- Pop a toolbar from the toolbar stack.

Argument list

```
METHOD FUNC$,TBL$[ALL]
```

FUNC$ Function:

"00" Define toolbar. You must define a toolbar before any other message structures can reference it. Defining a toolbar does not activate it. See 02.

"01" Modify toolbar. This allows you to redefine a toolbar. The arguments are identical to those required by the 00 function. If any of the arguments are left blank (except Toolbar name), the original values from the 00 function are used. This allows modification to a toolbar without having to duplicate all the arguments. For more information see TBL$[4].

"02" Activate toolbar.

"03" Deactivate toolbar.

"04" Pop toolbar.

TBL$[0] **Status.** Returns "." if the CALL to VIPTBAR was successful. Anything else indicates the CALL failed.

TBL$[1] **Toolbar name.** Required for all functions. This is the user defined name (alias) by which the host refers to the specific toolbar object. The maximum length is 8 characters.

TBL$[2] **Toolbar definition name.** Function 00 only. This is the name of the toolbar bitmap definition file on the workstation in the *vip-data-path* \ToolBar folder. The format for the toolbar name complies with the name format currently used by the Toolbar Maintenance module. The maximum length is 15 characters.

TBL$[3] **Toolbar title.** Function 00 and 01. This only applies when the toolbar position is F (Floating). The maximum length is 15 characters.

TBL$[4] **Toolbar options.**

(1,1) Initial Toolbar position. Functions 00 and 01. The default is R. When used with function 02, it overrides the original definition. If function 02 is used and the toolbar position is F, the toolbar position defined in the 00 and 01 function will be used.
F - Floating.
R - Docked right.
L - Docked left.
T - Docked top.
B - Docked bottom.

(2,1) Enable Float. Functions 00 and 01. The default is 2.

1 - User may undock the toolbar.

2 - User may not undock the toolbar.

(3,1) Enable Dock. Functions 00 and 01. The default is 2.

1 - User may dock the toolbar.

2 - User may not dock the toolbar.

(4,1) Enable ToolTip. Functions 00 and 01. The default is 1.

1 - Allow ToolTips.

2 - No ToolTips.

(5,1) Enable Hide Toolbar. Functions 00 and 01. The default is 0.

1 - User may hide the toolbar by clicking on the x box (upper right corner of the window) wile the toolbar is floating.

0 - User may not hide the toolbar.

(6,1) Toolbar Pop options. Function 04 only. The default is 1.

0 - When a toolbar is popped, the next toolbar in the stack remains hidden.

1 - When a toolbar is popped, the next toolbar in the stack is activated.

Example:

```
DIM TMSG$[4];                     !Dim toolbar array
LET FUNC$="00",                   !Set function to define a toolbar
TMSG$[1]="TBO",                   !Set local name for toolbar
TMSG$[2]="Toolbar2.def",          !Set definition name
TMSG$[3]="My Toolbar",            !Set toolbar title
TMSG$[4]="R" +                    !Set dock toolbar right +
   "1" +                          !Enable floating +
   "1" +                          !Enable docking +
   "1" +                          !Enable tooltips +
   "1"                            !Enable hide toolbar
                                  !Note: pop option not set, use
                                  !default ("1")
CALL"VIPTBAR",FUNC$,TMSG$[ALL]    !Call VIPTBAR to define (expose) the
                                  !toolbar
IF TMSG$[0]<>"."THEN
   TERMINATE                      !If not successful terminate
                                  !processing
ENDIF
LET FUNC$="02"                    !Set function to activate the toolbar
CALL"VIPTBAR",FUNC$,TMSG$[ALL]    !Call to activate the toolbar
IF TMSG$[0]<>"."THEN
   TERMINATE                      !If not successful, terminate
                                  !processing
ENDIF
CONNECT SCREEN 'OESCUST"          !Do connect screen
```

SYSTEM MESSAGES

Clear Input Buffer XG18

With type ahead, it is possible to have keystrokes buffered in the queue that are no longer appropriate for the application logic. This system message will clear the input buffer. Care should be taken when using this message in your application. The clearing of the buffer is indiscriminate; the buffer may contain keystrokes critical to your application logic. To clear the input buffer:

```
      CALL "VIPCOM","XG18"
```

Disable/Enable Keyboard and Mouse XG19

At times, it may be desirable to prevent the user from entering keystrokes or making selections with the mouse. This system message can be used to disable and enable both the keyboard and the mouse. Implementing VIP messages in your application code should always been done with care. Issue messages incorrectly or at an inappropriate point in application logic can create conflicts for Dictionary-IV and OPENworkshop.

```
      CALL "VIPCOM","XG190"    ! disable the keyboard and mouse.

      CALL "VIPCOM","XG191"    ! enable the keyboard and mouse.
```

OPENING WINDOW FILES

Window files and programs can be opened from a VIP or WorkStation Manager ToolBar or from a button on a custom graphical screen. The disadvantage these techniques is that the file must be predefined. Neither method provides for populating the file name dynamically at run time.

From within VIP, TbredComm can be used to open and execute external Window files such as image files, Word documents, Excel spread sheets, etc. Any file that can be opened or launched from the Windows Run dialog (Start button, Run) can be executed from within TbredComm.

For more information please see the TbredComm manual Opening External Window Files.

IMPORTING VIP SCREENS

The VIP Import Utility is provided to allow migration from VIP 4.0*x* to VIP 4.10.

- To move graphical screens to another host please see Screen Backup/Restore in this manual.
- To import graphical screens from VIP 4.0n to VIP 4.10 use the VIP Import Utility.

This is a one time upgrade process to allow you to use your existing VIP GUI screens. If you plan to generate all new GUI screens, there is no need to import your existing definitions. There are two steps to importing your existing VIP GUI screens to the host.

- Prepare the host environment for the import process.
- Import the screens and templates to Dictionary-IV.

Prepare Host Environment

The import utility requires both the VIP .frm files and the VIP .msg files. The .pos files are optional. If the corresponding .pos file can be opened, the saved location from the client will be saved with the GUI screen definition on the host.

Windows

In a VIP 4.0x installation these files can typically be found on the client in the following folders.

.frm files:
`C:\Tsi\ObjectCache\Screensg`

.msg and .pos files
`C:\Tsi\ObjectCache\Screens`

.frm files for templates
`C:\Tsi\ObjectCache\Screenst`

UNIX/Linux

The files must reside on the host. If you are running the Import Utility on a UNIX host, you must first move the files to the host. The recommended (not required) directories for a UNIX host are:

.frm files
`/usr/basic/Screensg`

.msg and .pos files
`/usr/basic/Screens`

Default Values

The Import Utility interface has default values for the location of your screen definitions. These default values are maintained in the Format UTFVSIM1.

The default value for the .frm files is maintained in UTFVSIM1.START-SCRN-NAME.

The default value for the .msg and .pos files is maintained in UTFVSIM1.SCRNS-PATH. To modify the default values to reflect your environment, edit this data element.

The following are example default values for UTFVSIM1.START-SCRN-NAME:

```
C:\Tsi\ObjectCache\Screensg
/usr/basic850/ObjectCache/Screensg
/usr/development/basic/Screensg
```

The following are example default values for UTFVSIM1.SCRNS-PATH:

```
C:\Tsi\ObjectCache\Screens
/usr/basic850/ObjectCache/Screens
/usr/development/basic/Screens
```

The same GUI Import Utility is used for importing VIP screen templates, however there are some special considerations. VIP templates are stored and managed in the Dictionary in the same way as VIP GUI data entry screens. For this reason, each GUI template must have a corresponding character definition and must reference the format definition UTFVT.

Rename or copy your templates in ObjectCache\Screenst to an 8-character Dictionary-IV compatible name. The language code and the file extension are not counted. For example:

Rename or Copy
DEFAULTTEMPLATE-EN.frm to UTVIP01-EN.frm

Move the templates from ObjectCache\Screenst on the client to the host. Template definition files are also .frm files and must be placed in the same directory with the data entry screen .frm files.

On the host select Dictionary-IV screen maintenance. Copy (**Do Not Rename**) the character screen UTSVS001 using the new 8-character name of your template. For example:

Copy
UTSVS001 to UTVIP01

In the above example the original template DEFAULTTEMPLATE-EN.frm will be maintained in Dictionary-IV as UTVIP01.

Importing Screens

Do Not turn on VIP, the import utility must be run in character mode. The import utility uses several system calls that are not currently supported by VIP.

From the OPENworkshop/Dictionary-IV Utilities Menu select the **VIP Utilities** option.

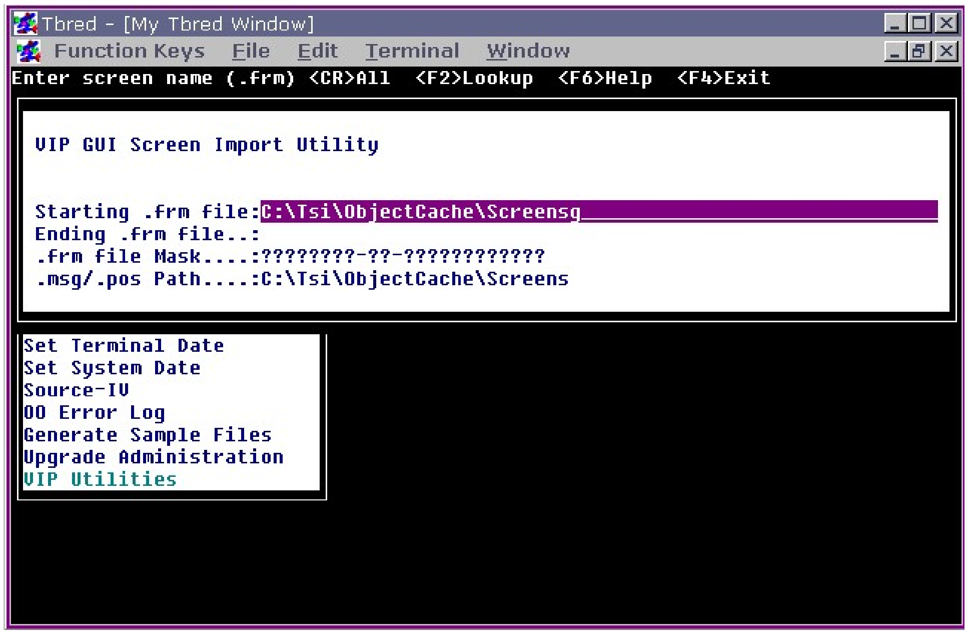
Select the **VIP GUI Screen Import** option.

You will be prompted:

```
Log file VIPSIML: Clear Append Exit
```

See below for more information on how to use this log file.

The VIP GUI Screen Import Utility screen displays. The following image illustrates running the import utility on a Windows host.



```
Enter screen name (.frm) <CR>All  <F2>Lookup  <F6>Help  <F4>Exit

 VIP GUI Screen Import Utility

 Starting .frm file:C:\Tsi\ObjectCache\Screensg
 Ending .frm file..:
 .frm file Mask....:????????-??-????????????
 .msg/.pos Path....:C:\Tsi\ObjectCache\Screens

Set Terminal Date
Set System Date
Source-IV
OO Error Log
Generate Sample Files
Upgrade Administration
VIP Utilities
```

Starting .frm File

Type the name of the first form file (VIP screen) to be imported and press **Enter**.

Press **Enter** to select all the .frm files in the specified directory.

Press **F2** for a LookUp of all the .frm files in the specified directory.

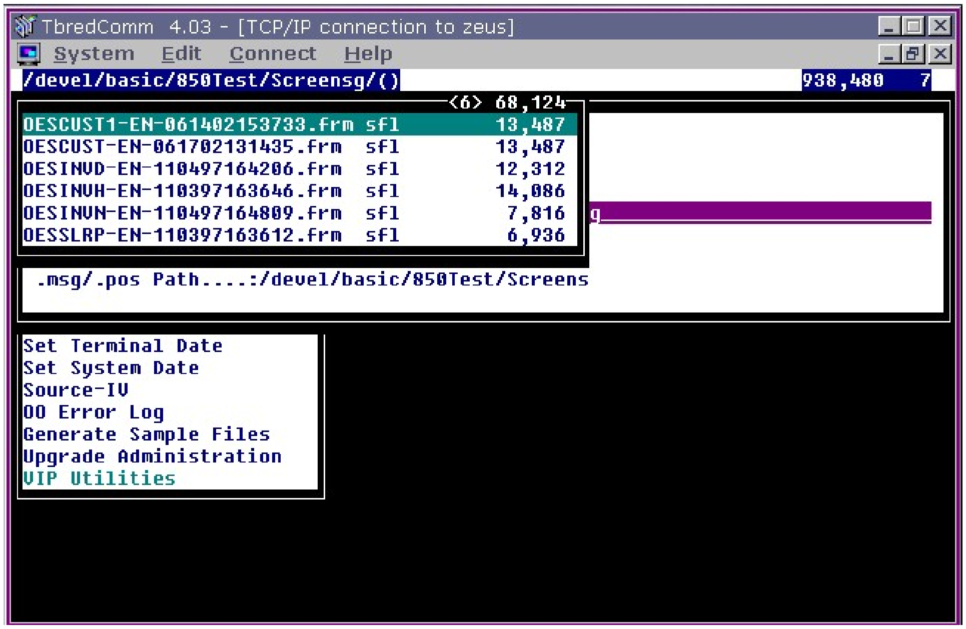
Press **F4** to exit the import utility.

If the screen frm file is not in a directory defined by a DIR entry in your IPLINPUT file, you must enter the fully qualified name.

The default directory is maintained in the Format definition UTFVSIM1 as a default value for START-SCRN-NAME. Edit this format modifying the default value to reflect your environment. For example:

```
C:\Tsi\ObjectCache\Screensg

/usr/basic850/ObjectCache/Screensg

/usr/development/basic/Screensg
```

The following illustrates the **F2** LookUp function on a UNIX host:



```
/devel/basic/850Test/Screensg/()                         938,480    7
                                   <6> 68,124
OESCUST1-EN-061402153733.frm sfl          13,487
OESCUST-EN-061702131435.frm  sfl          13,487
OESINVD-EN-110497164206.frm  sfl          12,312
OESINVH-EN-110397163646.frm  sfl          14,086
OESINVN-EN-110497164809.frm  sfl           7,816
OESSLRP-EN-110397163612.frm  sfl           6,936

 .msg/.pos Path....:/devel/basic/850Test/Screens

Set Terminal Date
Set System Date
Source-IV
OO Error Log
Generate Sample Files
Upgrade Administration
VIP Utilities
```

Ending .frm File

See "Starting .frm File."

./msg ./pos Path:

Type the directory where the .msg and .pos files are located on the host and press **Enter**.

Press **Enter** if these files are in a directory defined by the IPLINPUT file.

Press **F2** for a Lookup of available directories.

Press **F4** to exit the import utility.

The .pos files are optional. These contain the saved location of the GUI screen on the client. If a corresponding .pos file is found, that information will be stored with the GUI definition on the host.

If the screen .msg and .pos files are not in a directory defined by a DIR entry in your IPLINPUT file you must enter the fully qualified path where the .msg and .pos files are located.

The default directory is maintained in the Format definition UTFVSIM1 as a default value for SCRNS-PATH. Edit this format modifying the default value to reflect your environment.

During the import process the status of each GUI screen will be recorded in the VIP Import Log file. If a GUI screen fails to import, the log file will contain appropriate error messages. After all selected screens have been imported; the VIP Import Log View will be displayed.

VIP Import Log View

VIP Screen Name
The name of the imported VIP GUI screen definition.

Import Date
Date the VIP screen was imported to Dictionary-IV.

User Comments (Usr Cmt)
This text field is not used by the import utility. This field has been included for the developer to make any comments or notes regarding a particular screen. If you should select the option to 'Clear' the VIP Import Log, these comments will be lost.

Status
PASSED or FAILED.

Error Code
A list and explanation of error codes generated by the import utility.

Error Details
This is an occurrence field containing up to 5 entries. This provides detailed information regarding the error.

RELOADING LOCAL CACHE

The ObjectCache folders are used to store GUI definitions locally on each workstation. When a GUI object is requested by the host the client software will search local cache for that object. If it is found and if the timestamp for that object matches the host timestamp, the GUI object from local cache will be used. If the timestamps do not match a new GUI definition will either be created or the master definition will be downloaded from the host.

Performance can be improved by maintaining a master version of local cache and loading those directories on each workstation. VIP for Dictionary-IV includes a utility that will copy a mirror image of the ObjectCache folders to the *vip-data-path*\Tsi\ObjectCache directories. This utility can also be used to first empty ObjectCache and then reload it if local cache has become corrupt.

To reload ObjectCache run *vip-program-path*\TSSetupEnv.exe.

```
TSSetupEnv.exe -wash -load <load-directory-name>
```

-wash
When the -wash argument is supplied, TSSetupEnv.exe will clear the contents of:

> *vip-data-path*\Tsi\ObjectCache\Format
> *vip-data-path*\Tsi\ObjectCache\Help
> *vip-data-path*\Tsi\ObjectCache\Menu
> *vip-data-path*\Tsi\ObjectCache\Screens
> *vip-data-path*\Tsi\ObjectCache\Screensg
> *vip-data-path*\Tsi\ObjectCache\View
> *vip-data-path*\Tsi\ObjectCache\Viewsp with the exception of default.tpl.

-load
When the -load argument is supplied TSSetupEnv.exe will load the supplied files into the *vip-data-path*\Tsi\ObjectCache folders. The <*load-directory-name*> is optional. If not supplied, TSSetupEnv.exe will default to C:\vip-data-path\Tsi\LoadEnv as the source for the load environment. The directory structure of the load environment must mirror the entire directory structure of TSI\ObjectCache. For example:

> C:\861LoadEnv\ObjectCache\Format
> C:\861LoadEnv\ObjectCache\Help
> C:\861LoadEnv\ObjectCache\Menu
> C:\861LoadEnv\ObjectCache\Screens
> C:\861LoadEnv\ObjectCache\Screensg
> C:\861LoadEnv\ObjectCache\View
> C:\861LoadEnv\ObjectCache\Viewsp

VIP FAQ

VIP General

Can I use any terminal emulator, any terminal table?

VIP requires Thoroughbred's TbredComm terminal emulator when connecting to a remote host. In addition you must use the TBRECON terminal table. If you create a custom terminal table you must use "TBRED" as the first 5 characters of the terminal table name.

Can I use a Windows host?

VIP can run on a Windows network. Starting with 8.5.0 a Basic installed on Windows XP, Windows Server 2000, Windows Server 2003, Windows Vista, Windows Server 2008, and Windows 7.

VIP appears to time out or hang periodically?

General network problems over a slow network can often be corrected by reducing the maximum size of the inbound message: messages posted from the host to the client. If you experience general connectivity problems try reducing the Queue Size in the WorkStation Manager. Try reducing this to 16K or 8K. The default value is 16K; the maximum value is 64K

Why does VIP occasionally crash when displaying Screens or Views?

At times local cache can become corrupt. This might happen when the timestamps on the workstation and the host match but the GUI objects are not in sync with the host.

You can delete the problem GUI object from the appropriate *vip-data-path*\Tsi\ObjectCache folder or use `TSSetupEnv.exe` to reload local cache. See the Reloading Local Cache section of this manual for more information.

It is important when restoring IDDbk files that you also restore the corresponding IDDbkVIP file to keep the host character screen and GUI screen definitions in sync.

None of the VIP objects have the focus, which object is the current one?

Find Current Objects (**CTRL-F**)
The host controls what graphical object within the main VIP GUI Server window should have the focus. In a windows environment the user can click on any window displayed on the desktop. This inherent difference between host application code and a windows workstation paradigm can at times create a situation where the current VIP object cannot retain the focus. To find the current object press **CTRL-F**. Or from the VIP menu select "System" then select" Find Current Object".

How do I prevent my customers from incorrectly shutting down my application by closing the VIP or TbredComm window?

The WorkStation Manager contains options to disable the close button on both the main VIP GUI Server window and the TbredComm window.

I'm not using Desktop Application Menus, how do I have my application start with VIP enabled?

Please see VIPSTART discussed in the VIP for Dictionary-IV manual (Enable the Graphic Presentation).

Are there multiple spoken language versions of the VIP product suite?

No, however you can translate all the text presented by the VIP products; error messages, dialog boxes, menu bar text, etc. Please refer to the Multi Language for VIP for details.

Which fonts work well when migrating from character to GUI?

Always use fonts that are guaranteed to be installed on the client. If a font or point size is used by a VIP graphical object and that font is not installed the client, the OS will perform a font substitution. This may not yield the desired results. When creating GUI Screens, and setting default values in the WorkStation Manager, always use fonts that you know to be installed on each VIP client.

For proper data alignment in Screens and Views always use a fixed font. Keep in mind that the VIP interprets the character (80x24 grid) versions of the various Dictionary-IV Classes when creating the corresponding GUI objects. Resolution and font can have a dramatic impact on this interpretation. Consider fonts in a data entry screen. Assume an input field is defined as /FIELD(10)=data-element. VIP will calculate the width of that field based on 10 * the width of the W character in the specified font. By using fixed-width fonts you can avoid some of the slight variations that come into play with this averaging. Recommended fonts for on-line help are DejaVu Mono, Courier New, and IBMPCDOS. For more information please see refer to the WorkStation Manager manual (Screen, Menu and Help Fonts tab).

On-line Help will be displayed using the font defined in the WorkStation Manager. If your help contains business graphic characters you must use a font and terminal table that will support these characters. When editing help text with VIP enabled it is critical that the column and row mapping for the character presentation be preserved.

NOTE: The Tbred font is not supported on Windows Vista, Windows Server 2008, and Windows 7.

Why does VIP occasionally crash when displaying on line help?

Help generated and cached on the client using VIP 4.031 or earlier is not compatible with VIP 4.10. To correct this problem, clear the contents of the TSI\ObjectCache\Help folder. In release VIP 4.10 the communication of help was modified in order to correct several problems making it incompatible with cached help definitions created by prior releases.

Views

How do I control where the GUI objects appear within the VIP GUI Server window?

The starting location of a Screen or a View in character mode may not be the best starting location for that same object in GUI mode. In the WorkStation Manager there is an option to save the location of a GUI object where it was last displayed. For example, col 5 might make sense in char mode for the start col of a view but may not be the optimal location running with VIP enabled. It is important to remember that a location running in one resolution might be inappropriate in another resolution.

In addition GUI objects typically require less 'real estate' on the desktop than the corresponding character definitions. Experiment with resizing and saving GUI object locations on the client to optimize the presentation of your application.

How to I control data alignment in a View?

As mentioned above, fixed-width fonts will always produce the best alignment of data in a column. Recommended fonts are DejaVu Mono, Courier New, and IBMPCDOS. For more information please see refer to the WorkStation Manager manual (Screen, Menu and Help Fonts tab).

In addition the View Rules setting in the WorkStation Manager will control how data is aligned within a View. By enabling this option all date fields, all numeric fields and all fields where the padding types are 1 or 3 will be right justified. All other field types and padding types will be left justified.

What GUI presentation is used for the View Pre/Post Processing options?

A right mouse click context menu is automatically created for each column based on the format attributes of the data element. These options include: Lookup, Connects, Help and Properties. If a 'class' of options is not defined it will appear as disabled in the context menu. The 'Properties' option is only available when in developer mode. This is the equivalent of doing an F11 to modify the format attributes of the data element.

Tool Bars

Does VIP support a Tool Bar?

The VIP product supports two Tool Bars: the WorkStation Manager Tool Bar that is tied to the desktop and the VIP Tool Bar that is contained by the main VIP GUI Server window. Consider using a VIP Tool Bar for those functions that are common to the various object classes in your application. For example: LookUps, Exit, Help, etc. The VIP Tool Bar is dynamic. It is possible to define multiple Tool Bars and control, which Tool Bar is on top of the stack at any point in your application code.

Creating GUI Screens

NOTE: The VIP GUI Screen Editor is not supported on Windows Vista, Windows Server 2008, and Windows 7. Please use a Windows XP, Windows Server 2000, or Windows Server 2003 system for editing your GUI screens.

What do I need to know when creating and editing GUI screen definitions?

All screens start as default screens. Based on the character definition VIP will create a graphical interpretation of the screen definition. You can control some of the rules that are applied to this logic by configuring the Screen Filter File. For example using this filter you can control if the field label “Customer Code” will be created as two controls or one. Configuring this file appropriately will produce default screens that are easier to modify and change in the Graphic Screen Editor.

Before converting all of your screen definitions pick a few screens that represent a good cross sample of your application and experiment with a few different styles. You can create several different templates and apply them in the GUI screen editor to get a quick feel for how your application might look. Converting all your screens and getting them all just right can be a fair amount of work. Taking the time up front to work out a template and a style will reduce the effort of converting all your screens.

For input fields stick with fixed-width fonts. Recommended fonts for on-line help are DejaVu Mono, Courier New, and IBMPCDOS. For more information please see refer to the WorkStation Manager manual (Screen, Menu and Help Fonts tab).

If you will be adding buttons to your screens, use the option in the Graphic Screen editor to define default values for your buttons. Adding buttons in this way will provide consistent font attributes and a consistent size to all your buttons across all your screens. Also consider using a VIP Tool Bar for functions that are common across your application. This will provide a clean easy to read screen and a common interface for your user.

How do I select multiple controls to move or align a group of controls?

First be sure that the Shift Down button is not enabled. The Shift Down button is a shortcut to the Properties dialog and will take precedence over making multiple selections. While holding down the **Ctrl** key, use the mouse click to select multiple controls. Each selected control will be highlighted with grab handles. Once a group of controls are selected, the entire group and be moved simply by dragging the group or aligned using one of the alignment options on the Editor Tool Bar.

Why do controls 'jump' when I move them in the editor?

When the option for "snap to grid" is enabled controls will be moved to the nearest point on the grid. Experiment with disabling the grid control in the Graphic Screen Editor and using the various Align options on a selected group of controls. It is a matter of preference however some users find it easier to work with the grid disabled.

Why do I get the warning 'Caption and Tips cannot contain more than 50 characters'?

There is a limit to the length of text that can be used for text (caption) and tool tips. This limit is not strictly tied to these two properties alone. Specifically there is a limit to the total number of characters used to store the definition of a control. On average there are approximately 50 characters available for both the text and tool tips.

VIP appears to time out or hang when uploading screen definitions?

When working with large custom GUI screen definitions over a slower network it may be necessary to adjust the maximum size of an outbound message: messages posted to the host. If you experience problems saving large GUI screen definitions back to the host try reducing the Upload Message Size in the WorkStation Manager.

In addition to network speed the client hardware and operating system can contribute to problems when editing large GUI screen definitions. Most current hardware and operating systems can run with an Upload Message Size of 64K. On older machines the Upload Message Size may have to be reduced to 8K. In extreme cases it may not be possible to edit large graphical screens. For editing large GUI screens we recommend at least a P2 with a minimum of 128MB RAM and Windows Server 2000 or Windows XP.

Why is a template being applied to a screen when it is loaded in the editor?

This can happen for one of two reasons:

In the Graphic Screen Editor under Edit, Options, Templates there is an option to automatically apply the selected template when a screen is loaded in the editor.

From Dictionary-IV Screen Maintenance when you select a screen for editing and there is no corresponding GUI screen, a default screen will be created for the editor. If in the WorkStation Manager the to option to apply a template to a default screen is enabled, that template will be applied to the default screen created for the editor.

Why isn’t the WorkStation Manager (Screen Template Options) not showing the current list of templates?

The list of templates displayed by the WorkStation Manager is updated only when a GUI screen template is loaded in the GUI screen editor. Loading a template in the editor is the only way to update the client with a template definition. If a template definition is not included in the list, simply load that template in the GUI screen editor to update the client with the template definition.

Why is the screen loaded in the editor when I do a CONNECT SCREEN?

It is recommended that the Graphical Screen Designer be closed before trying to access screens at run time. The screen designer is not closed after a screen definition is uploaded or unloaded. This is done to avoid having to reload the designer each time a GUI screen is selected for editing by the host. If the designer is not closed and a CONNECT SCREEN or INPUT SCREEN is requested, under some circumstances the screen will incorrectly be loaded in the editor.

Run Time GUI Screens

The foreground and background colors of certain screen controls are difficult to read, how do I correct this?

You can control the foreground and background color for Security fields, Calculated (formula) fields, and the current field in the WorkStation Manager. These foreground and background colors are applied to the control when the GUI screen is rendered at run time. When selecting colors, keep in mind that the OS will apply the system highlight color to the active field. When selecting default colors for the current field, consider using a color combination that will not conflict with the system highlight colors.

Why has VIP slowed down when creating GUI screens?

Never run with the OCX Message trace enabled unless you have been requested to do so by Customer Support. This will dramatically reduce the performance of screen processing. Don't forget to disable message tracing in the WorkStation Manager if you have enabled it to generate trace files for Customer Support.

Why are the default screens very large and empty or corrupted?

This can happen for one of two reasons.

- The ActiveX control (ScrControl.ocx) used for processing screens has either been renamed or moved causing the registry entry to be altered. To correct this, please refer to registering the OCX control using TSTCON32.exe described in the VIP for Dictionary-IV manual.

- The default font or point size for Screens defined in the WorkStation Manager is not installed on the client. Also see 'Why is the font used by the default screens so small?'

Why is the font used by the default screens so small?

This is a problem with the way Windows handles font selection. The Windows Common Dialog for selecting fonts displays font sizes that are not necessarily available for the selected font type. Windows attempts to match the closest font size, which can yield unexpected results. Another related item that contributes to the font confusion is a difference in the way fonts are handled in Windows and they way they are handled in Visual Basic. Depending upon the font, there can be some variation in point size.

Recommended fonts are DejaVu Mono point size 11 or Courier New point size 11.

Slave Printing

Why do slave print jobs remain the windows spooler?

You must PRINT the ‘PC’ mnemonic before closing the slave printer. For example:

```
0010 OPEN(1)"UTCUST;
     OPEN(2)"PS";
     PRINT(2)'OPEN'

0020 K$=KEY(1);
     READ RECORD(1)A$;
     PRINT(2)A$

0030 PRINT(2) 'CLOSE',;
     PRINT 'PC',;
     CLOSE(2)

0090 END
```

TbredComm

What is the largest font I can use with TbredComm?

The font size is dependent upon the font used. For example Courier New supports a point size of 72.

WorkStation Manager

I disabled the WorkStation Manger icon in the system tray, how do I get it back?

Once the option to show the WorkStation Manager icon in the system tray has been disabled, it can only be enabled by manually editing the Tbred.ini file. Using Notepad or any text file editor open *vip-system-path*\Tbred.ini in the [TWMGR] section change TrayIcon=no to TrayIcon=yes and reboot the client.

How do I prevent my customers from altering the Option settings in the WorkStation Manager or from selecting certian items available on the WorkStation Manager menu?

There are two ways to accomplish this.

- You can prevent the WorkStation Manager icon from being displayed in the system tray. Set the ‘Tray Icon’ option on the WorkStation Manager tab to ‘No’. Please refer to the WorkStation Manager manual for details.

- Individual items can be deleted from the WorkStation Manager menu. Please refer to this faq How Can I Customize the WorkStation Manager Menu?

How can I Customize the WorkStation Manager Menu?

The context menu that is displayed when doing a right mouse click on the WorkStation Manager icon can be modified. Entries can be deleted, text can be altered but entries cannot be added. The content of the menu is defined by the file OOMGRMNU-*language_code-timestamp* located in the *vip-data-path* \Tsi\TWMgr folder. Note that this file is not downloaded from the host. Changes to this file must be made on each client.

For more information please refer to the Multi-Language for VIP manual.

In some cases it is preferred to remove one or more WorkStation Menu items. This is a two-step process. The OOMGRMNU file is modified, removing the desired entries and the noitems= in the [TWMGR] section of the Tbred.ini file is updated to reflect the new menu item count. Both of these files can be edited using any standard text editor, for example notepad.exe.

NOTE: The first entry in the file contains the tool tip text; this is not a menu item.

For example, the "Menus" and "Applications" items can be removed from the WorkStation Manager menu. OOMGRMNU before deleting the "Menus" and "Applications" entries:

```
-1MWorkstation Manager Menu[]
00XOptions[40075]
00MToolbar[]
01MToolbar Position[]
02SNone[40057]
02SFloating[40058]
02STop[40059]
02SBottom[40060]
02SLeft[40061]
02SRight[40062]
01XLoad New Toolbar[40072]
01XToolbar Maintenance[40071]
00MMenus[]
01XRemove Application Menu[40068]
01XRestore Application Menu[40073]
00MApplications[50004]
00MHost Type[]
01SRemote Host[40063]
01SWorkstation[40082]
00MPresentation Mode[]
01SCharacter[40065]
01SVIP[40066]
00XStop Manager[40070]
```

Tbred.ini before deleting the ‘Menus’ and ‘Applications’ entries:

```
[TWMGR]
noitems=22
```

OOMGRMNU after deleting the ‘Menus’ and ‘Applications’ entries:

```
-1MWorkstation Manager Menu[]
00XOptions[40075]
00MToolbar[]
01MToolbar Position[]
02SNone[40057]
02SFloating[40058]
02STop[40059]
02SBottom[40060]
02SLeft[40061]
02SRight[40062]
01XLoad New Toolbar[40072]
01XToolbar Maintenance[40071]
00MHost Type[]
01SRemote Host[40063]
01SWorkstation[40082]
00MPresentation Mode[]
01SCharacter[40065]
01SVIP[40066]
00XStop Manager[40070]
```

Tbred.ini after deleting the ‘Menus’ and ‘Applications’ entries:

```
[TWMGR]
noitems=18
```

MULTI-LANGUAGE

For more information about Multi-Language, see the Dialog, Message, and MenuBar Text Translations section and the Character and Font Translation Table Files section in the Multi-Language for VIP.