The New *Macro Edit* in AnzioWin version 12.6

What's New in Version 12.6

With the release of AnzioWin version 12.6 we are introducing a new interface supporting the editing of user-defined macros. Prior releases of AnzioWin utilized a "function-line" approach to editing the user macros.

With this release of version 12.6, we have added a complete visual interface, with plans to further enhance it as we get feedback from users.

Introduction To AnzioWin Macros

AnzioWin can associate a user-defined string (macro) with either a special key in the key file, or with a single character off of the keyboard. These macros can then be initiated on startup, from the keyboard or from a host command.

Macros built around a special key are immediately accessible by pressing that key. Those defined as a character are accessible either by invoking them n start up, from another macro, from the host or from the "Transmit:" key line (available by pressing ctrl-F1 or shift-Ctrl-F1). The keys can be strings to send to the host, terminal emulation escape sequences, commands for AnzioWin, or local displays that have no effect on the communications channel.

Note: Macros should not be confused with AnzioWin's scripting capabilities. Macros are quick and easy key definitions that are designed to aid the user. In fact, you usually will set up a macro to run a script.

The macros are stored in the key mapping file, similar to other key sequences. The name of the key file is carried in your current defaults file, making it available to that default file whenever you start up AnzioWin.

Rasmussen Software, Inc.
10240 SW Nimbus Ave., Suite L9, Portland, Oregon 97223 USA

The New Interface

In the past, macros were entered through AnzioWin's Function line as a "define" command. In version 12.6, we have introduced a few new menu items under the Macros menu.
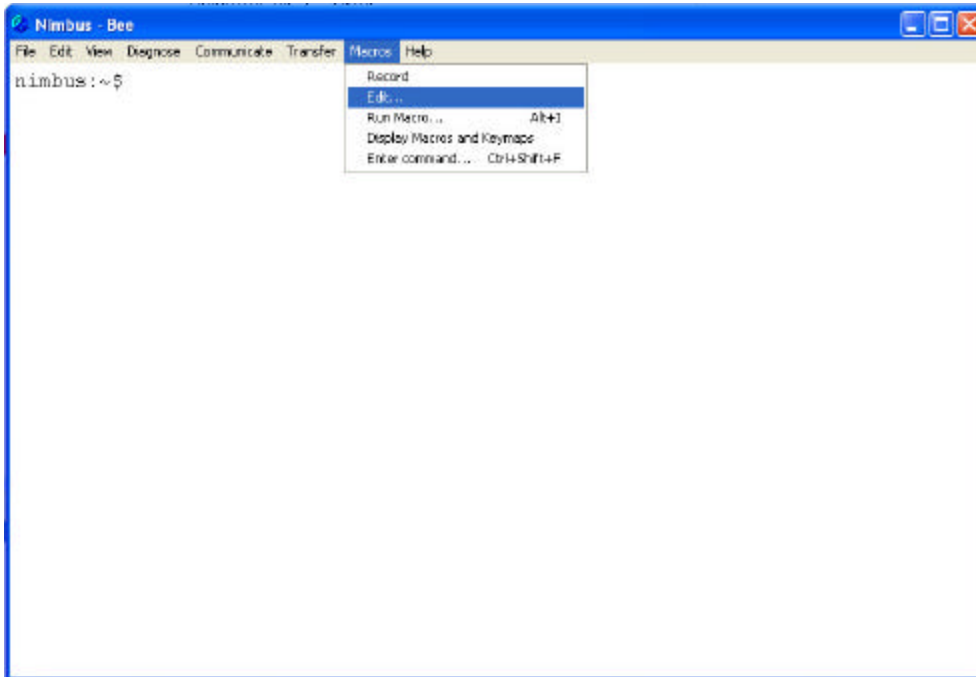
From the Macro menu, you can Record a macro, Edit a macro, Run a, list all macros or jump to AnzioWin's Function line.

Record

Record allows you to turn on the record feature in AnzioWin and begin a recording session (indicated by a check mark). While recording, all keystrokes will be stored and can then be played back later. Once you are done recording, simply come back the this menu item and click on it. You will then be asked what letter or special key you wish to save it to.

Edit

We will deal with the editing of macros in the rest of this document (see below).

Run

When you select the Run menu item, you are taken to the "Transit Key:" line in AnzioWin (Ctrl-shift-f1 will do the same). Here you can enter the special key or the single character that represents the macro you wish to run. If you have matched a special key, such as a function key, to a macro, you do not need to go here to run the macro, you can do so by simply pressing that keystroke on the keyboard. For other characters, use this feature to run the macro.
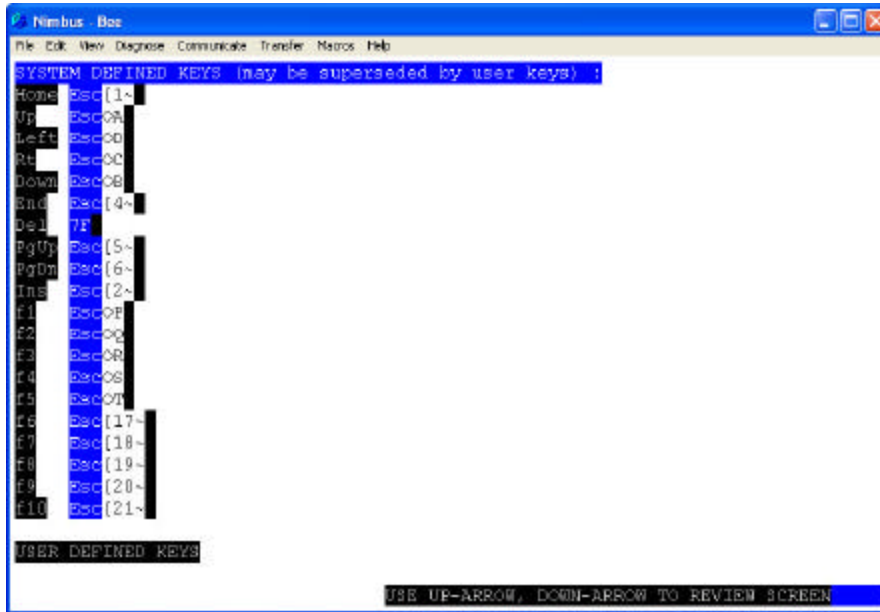
Display Macros and Keymaps
When you select this menu item, a list of the keys defined in the key file is displayed. This list is separated into two parts, "System Defined Keys" and "User Defined Keys".

The "System Defined Keys" represent those keys defined for the specific terminal type you are running, i.e. Wyse50 will have a different key map than will SCO ANSI or vt220.
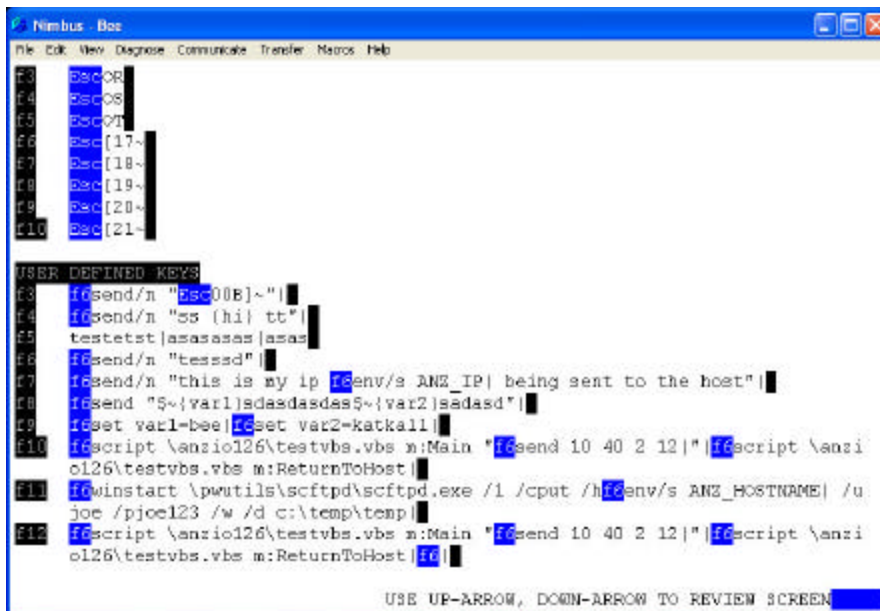

The "User Defined Keys" are those which the user has defined. These may override the current terminal emulation definition for a key defined above.

Rasmussen Software, Inc.
**10240 SW Nimbus Ave., Suite L9, Portland, Oregon 97223 USA**

Note: When AnzioWin displays keys here, it represents the defined key in one reverse video color and special keys in the definition in another reverse video color. This means the actual key was pressed. There are a couple of exceptions, including a carriage return. Carriage returns are represented as a vertical bar or pipe (|). Further information on the layout of a macro can be found below or in the documentation on AnzioWin.

Enter Command
When you select this menu item, you are taken to the Function line within AnzioWin. The Function line allows you to manually enter commands for AnzioWin, just as they might come in from a host script, PC script or a macro.

The Edit Macro Window
The Edit window allows you to bring up and edit the definition for a macro.
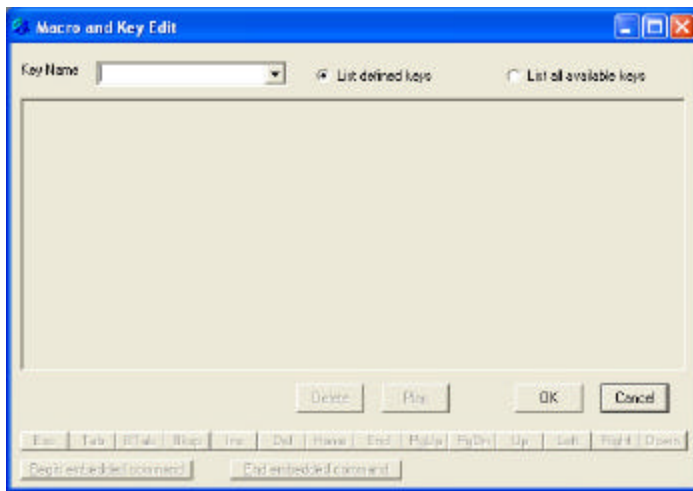
The screen is separated into a Key Name, the Definition field, a Delete, Play, OK and Cancel button, and the Special Keys buttons.

Key Name Field
The Key Name is the key representing the macro. This may be a special key, such as a Function key (entered by actually pushing that key when in this field), or a letter, such as a or A or L, used to represent the macro. This field is case sensitive.
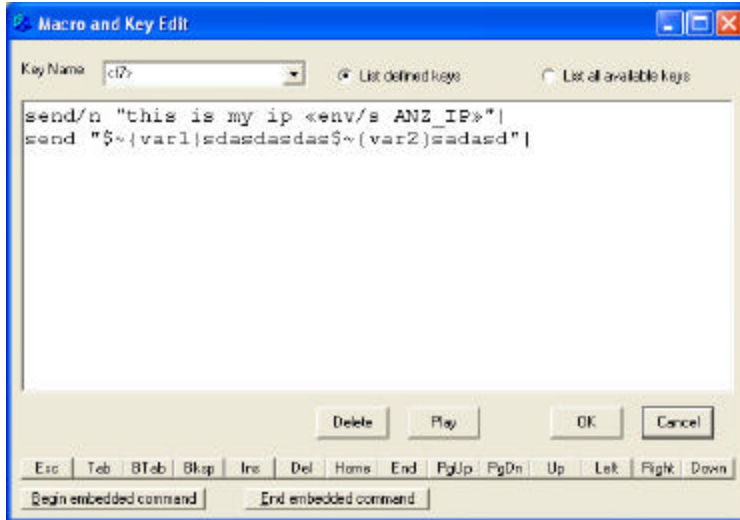
When you press a key here or select from the list, if a macro is defined, the definition of the key will be shown.

List defined keys / List all available keys
This option lets you toggle the scroll list to show only defined macro keys, or the defined keys and all possible keys available to define. Select one or the other. By default, only defined macros are shown.

The Definition Field

The Definition field is where you can view a key's definition and edit it as necessary. We will cover this in depth below.

Delete Button

The Delete button is used to undefine a key. If you have previously set up a macro for a key, use this button to clear it (the key may still show in the Key Name list, but using that key name will have no affect).

Play Button

The Play button will close the Edit window and begin a run of the macro that is currently displayed.

Ok Button

By pressing the Ok button, the current key is saved (if any change took place) and the dialog is exited.

Cancel Button

When you press cancel, if any changes have occurred to the current key, the changes are discarded and the dialog exited.

Special Keys Buttons

These buttons, located across the bottom of the screen, allow the user to enter special keystrokes into the macro that might otherwise not be allowed. An examples would be entering arrow keys or the page-up. These keys are used in the edit field and can not be entered by pressing them on the keyboard.

The "Begin Embedded Command" and "End Embedded Command" buttons are for entering a command that is meant for AnzioWin within a macro (a command within a command). These are represented with special characters (begin is "«" and end is "»"). See some of the examples below.

When Keys Get Updated

AnzioWin loads the current key file into memory on startup, holding it there for quicker access. When you edit a key with this dialog, it overrides what is in memory, NOT what is stored in the key file. If you make any changes to a macro or key definition, you will be asked upon exiting if you wish to "Save keys". Answer yes to store what is in memory out to the key file. You can also do this under the File menu and "Save Keys".

The macro edit dialog will update the memory definition when any one of the following takes place:

- the Delete button is pressed
- the Play button is pressed and the definition field has changed
- the Ok button is pressed and the definition field has changed
- a different key is selected in the Field Name box and the definition field has changed (the current key is saved to memory before the next is displayed)

The Definition Field

Now for the Definition field. This component of the Edit dialog, allows you to enter and edit a macro's definition. This is a standard text edit box with many controls available to make editing easy.

Editing

Here is a list of the special keys used in this field to aid in adding and editing a macro.

Backspace	Deletes the prior character.
Delete	Deletes the next character.
Home	Goes to home position of current line
Ctrl-Home	Goes to first position of macro
End	Goes to last position of current line
Ctrl-End	Goes to last position of macro
PageUp	Goes up a page
Ctrl-PgUp	Goes to 1st position on current page
PageDown	Goes down a page
Ctrl-PgDn	Goes to last position on current page
Insert	Toggles overtyping / insert mode
Tab	Tabs out of this field to the next
Ctrl-shift-Num5	Select all
Ctrl-shift-Home	Select current position to begin of macro
Ctrl-shift-End	Select current position to end of macro
Ctrl-shift-PgUp	Select current position to begin page
Ctrl-shift-PgDn	Select current position to end page
Ctrl-shift-Left-Arrow	Select prior word
Ctrl-shift-Right-Arrow	Select next work

Shift-Up-Arrow	Select to begin of line
Shift-Down-Arrow	Select to end of line
Shift-Left-Arrow	Select prior character
Shift-Right-Arrow	Select next character
Ctrl-Z	Undo
Enter	Start new line

These are listed here partly to let you know they exist, but also to let you know that you need to use a different method of adding them to a macro. This is the reason for the special keys located at the bottom of the dialog.

As an example, to enter a Ctrl-PgDn you would hold down the control key and press the PgDn button at the bottom.

What is a Command / What is a Line?
In the definition field, each line is a command destined for AnzioWin. These commands, normally entered at the Function line in AnzioWin, can be done quickly from a macro by simply entering the command. Each line is then terminated with a carriage return (represented by a "|"). However, within a command to AnzioWin can be embedded other commands for AnzioWin (hence the embedded command character sequence).

Special key and character entry
Most any keyboard character can be entered within the definition field, just as with any other editor. However, as mentioned above, certain key combination are used in editing. Also there is the issue of entering key combination for foreign characters, such as combining diacritics.

When you need to enter an editing key or key combination, you do so using the Special Key Buttons found at the bottom of the screen. As an example, a PageUp normally takes you up a page within the editing field, whereas a Ctrl-PageUp takes you to the first character of the current displayed page. So in order to enter these in a macro to send to the host, you would press the "PgUp" button at the bottom of the page (this should show as "<PgUp>" within the edit field) and hold down the Control and press the "PgUp" button for Ctrl-PageUp (this should show as "<^Pup>").

Foreign characters can be entered in the edit field also. Foreign characters are entered by their Unicode decimal value from the numeric key pad by holding down the Alt key and entering the numeric value for the character (num-lock may need to be on for this to work and may not work under older versions of Windows). As an example to enter the CJK character represented by hex 559E, you would enter the decimal equivalent of 21918 and the character ? should appear.

AnzioWin also supports entering of combining diacritics. This is done using a deadkey entry (the first key does nothing, the second key pressed gets combined with the diacritic). The approach used here is similar to Microsoft Word. The combining diacritic special keys are entered first as a Control- and then the character below, followed by the

character to combine with. As an example, to enter ñ we would do a Ctrl-~ (ctrl-shift-`) and then 'n'.

Note that only certain combinations work, refer to the AnzioWin documentation for more information.

`	grave
"	acute
^ or >	hat
~	tilde
-	macron
(	breve
*	dot
: or '	umlaut
?	hook above
<	hacek
,	cedilla
.	dot below
_	underline
=	double underline

Notes on editing

- A vertical bar, "|" is used to represent a carriage return.

- A combination of chevrons, « » , represents an embedded command for AnzioWin to execute. This allows you to embed AnzioWin commands inside other AnzioWin commands.

- A combination of greater-than/less-than, < >, may represent an embedded special character, such as a function key or one of the edit keys (PgUp, Home, End, …). As an example, a Ctrl-PgDn would appear as <^PDn> while an F10 would show as <f10>. If AnzioWin cannot interpret the special key name, then it leaves these alone. In this manner you could enter tagged code, such as for Print Wizard's mark-up language within the macro.

- A tilde, ~, is used in those cases where AnzioWin should not interpret the next character. As an example, if you need the macro to send a vertical bar, you would represent it as "~|" and to send a greater than you would do "~>".

- A pound sign, #, represents where a pause should take place. When a "#" is encountered on a line by itself, the macro pauses, waiting user input followed by a return or enter.

- A combination of curly brackets, {…}, surrounding any text on a line by itself, represents something to be displayed locally on the screen, but the characters are not

Rasmussen Software, Inc.
10240 SW Nimbus Ave., Suite L9, Portland, Oregon 97223 USA

sent to the host.

- You can undefine a macro by deleting it. However, if the key combination has a default, such as may be defined for the terminal type, then it reverts back to that definition. If you want to mark a key as having NO value, such as to disable an accelerator key, you would set the value to {} with no text between

Notes On This Release

- In the past, a macro that sent text to the host, would simply show as the text. With a move here to a more controlled command structure, characters to be sent to the host are sent using the
  Send/n "text"
  command. When the Send/n or the Send command includes a string enclosed in quotes, that string is sent to the host as is. Send/N does not send a terminating carriage return, while Send does. We use Send/n when translating old macros to the new format, hence assuming a carriage return is present already.

- Conversion of existing macros is done at the time they are displayed and edited. No macros are converted unless you call them up and force an update. Old format macros should run just fine.

- You can always verify what you've done by looking through the format displayed by using the "Display Macros and Keymaps" menu item.

- If all else fails or you are not getting desired results, you can still edit macros the old way. Check out our web site Support pages for editing AnzioWin using the "Define" command.