



# Entire Screen Builder

---

Version 5.2.1

Basic Rules Reference

This document applies to Entire Screen Builder Version 5.2.1 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

© Copyright Software AG 1999-2003

All rights reserved.

The name Software AG and/or all Software AG product names are either trademarks or registered trademarks of Software AG. Other company and product names mentioned herein may be trademarks of their respective owners.

# Table of Contents

<b>Basic Rules Reference</b>	1
Basic Rules Reference	1
<b>3270 Keys Toolbar</b>	3
3270 Keys Toolbar	3
Maintaining the 3270 Keys Toolbar Rule	3
Overview of Options	4
Defining An Image for a 3270 Keys Toolbar Button	5
<b>BS2000 Keys Toolbar</b>	7
BS2000 Keys Toolbar	7
Maintaining the BS2000 Keys Toolbar Rule	7
Overview of Options	8
Defining An Image for a BS2000 Keys Toolbar Button	9
<b>Buttons for Main Dialogs</b>	11
Buttons for Main Dialogs	11
Maintaining the Buttons for Main Dialogs Rule	11
Overview of Options	12
Defining the Properties for a Push Button	14
Type	14
Styles	18
Mouse	20
<b>Buttons for Child Dialogs</b>	21
Buttons for Child Dialogs	21
Maintaining the Buttons for Child Dialogs Rule	21
Overview of Options	22
Defining the Properties for a Push Button	23
<b>Child Window</b>	24
Child Window	24
Maintaining the Child Window Rules	24
Overview of Options	26
<b>Control Attributes</b>	28
Control Attributes	28
Maintaining the Control Attributes Rule	28
Overview of Options	29
<b>Delete Characters</b>	31
Delete Characters	31
<b>Delete Lines</b>	33
Delete Lines	33
<b>Delete Prompt</b>	34
Delete Prompt	34
Maintaining the Delete Prompt Rules	34
Overview of Options	36
<b>Font and Colors for Main Dialogs</b>	37
Font and Colors for Main Dialogs	37
<b>Font and Colors for Child Dialogs</b>	40
Font and Colors for Child Dialogs	40
<b>Frames</b>	42
Frames	42

<b>Function Keys</b>	44
Function Keys	44
Maintaining the Function Keys Rules	45
Overview of Options	47
Defining the Detection Rule for a Specific Function Key	48
<b>Function Keys Toolbar</b>	50
Function Keys Toolbar	50
Maintaining the Function Keys Toolbar Rule	50
Overview of Options	51
Defining An Image for a Function Keys Toolbar Button	52
<b>Group Box</b>	54
Group Box	54
Maintaining the Group Box Rules	54
Overview of Options	58
<b>Image</b>	60
Image	60
Maintaining the Image Rules	60
Overview of Options	62
Defining Constant and Dynamic Images	63
<b>Item</b>	65
Item	65
Maintaining the Item Rules	65
Overview of Options	67
<b>Lines</b>	69
Lines	69
Maintaining the Lines Rules	69
Overview of Options	71
Examples	72
<b>Map Detection</b>	73
Map Detection	73
Maintaining the Map Detection Rules	73
Overview of Options	75
General	75
Regions	76
<b>Message Line</b>	78
Message Line	78
Maintaining the Message Line Rule	78
Overview of Options	80
<b>User Exit</b>	82
User Exit	82

# Basic Rules Reference

This documentation provides detailed information about the following basic rules.

- 3270 Keys Toolbar      Display a toolbar with buttons for the 3270 special function keys (such as SysReq or PA1).
- BS2000 Keys Toolbar    Display a toolbar with buttons for the BS2000 special function keys (F keys, K keys, DUE1 and DUE2 keys).
- Buttons for Main Dialogs    Create push buttons which correspond to PF keys, 3270 keys or BS2000 keys for the main dialogs.
- Buttons for Child Dialogs    Create push buttons which correspond to PF keys, 3270 keys or BS2000 keys for the child dialogs.
- Child Window            Convert the popup windows in the character screens to child windows.
- Control Attributes        Define the control size and special characters used in your legacy application, and define VGA screen support.
- Delete Characters        Define the characters that are not to be displayed.
- Delete Lines            Define the lines that are not to be displayed.
- Delete Prompt            Define the prompts that are not to be displayed.
- Font and Colors for Main Dialogs    Define the background color/image for the main dialogs, and the font, style, size and/or color for the following attributes: normal, underline, intensified, blinking and reverse.
- Font and Colors for Child Dialogs    Define the background color/image for the child dialogs, and the font, style, size and/or color for the following attributes: normal, underline, intensified, blinking and reverse.
- Frames                  Define up to four dialogs for which extended rules have been defined. These dialogs can be placed at the top, bottom, left and right of the dialog containing the basic rules.
- Function Keys            Define the detection logic for the function keys in the character screens.
- Function Keys Toolbar    Display a toolbar with buttons for host keys.
- Group Box                Convert boxes in the character screens to group boxes.
- Image                    Display images in the dialog.
- Item                     Move or copy text from the character screen to the title bar or status bar.
- Lines                    Convert lines in the character screens to Windows lines.
- Map Detection            Identify a screen in which rules are to be applied in map scope.
- Message Line            Display the message line in the status bar or in a specific line of the dialog.
- User Exit                Define a DLL file containing user exits.

See the documentation *Defining the Rules Using the SDK* for further information.

# 3270 Keys Toolbar

Use this basic rule to display a toolbar with buttons for the 3270 special function keys (Clear, Att, Reset, SysReq, PA1, PA2 or PA3).

This chapter covers the following topics:

- Maintaining the 3270 Keys Toolbar Rule
- Overview of Options
- Defining An Image for a 3270 Keys Toolbar Button

See also: *3270 Keys Toolbar* in the *GUI Viewers* documentation.

---

## Maintaining the 3270 Keys Toolbar Rule

For each button that is to be shown in the 3270 keys toolbar, you can either define an image or a default icon.

The default icons are shipped with Entire Screen Builder. They are provided as Windows resources in a DLL file. It is not possible to edit or change the default icons.

### To display all defined toolbar buttons

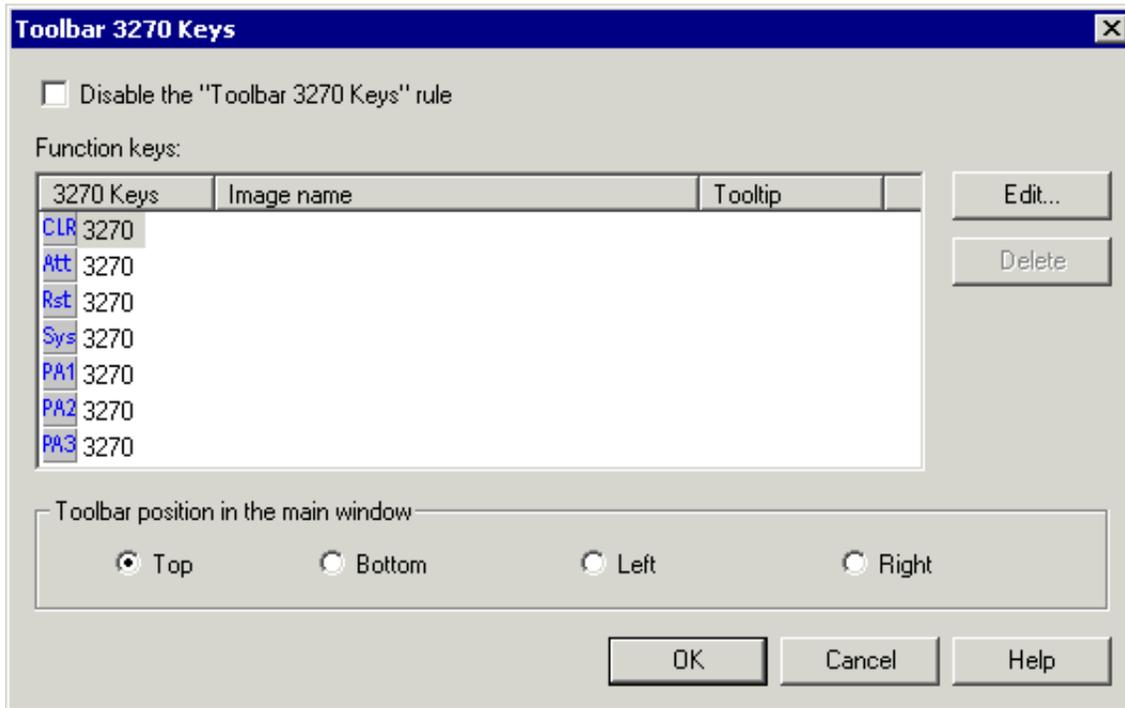
- From the **Basic** menu, choose **Toolbars > 3270 Keys**.

The Toolbar 3270 Keys dialog box appears. When images and tooltips have already been defined for a toolbar button, this is shown in this dialog box.

The following command buttons are available:

<b>Edit</b>	Modify the toolbar button definition for the selected 3270 key. Alternative: double-click a 3270 key. See <i>Defining An Image for a 3270 Keys Toolbar Button</i> .
<b>Delete</b>	Delete the toolbar button definition for the selected 3270 key. The 3270 key itself is not deleted.

## Overview of Options



### Disable the Toolbar 3270 Keys rule

When this check box is selected, the 3270 key toolbar is not shown.

### Toolbar position in the main window

Select an option button to define the position in the window (top, bottom, left or right) at which the toolbar is to be shown.

## Defining An Image for a 3270 Keys Toolbar Button

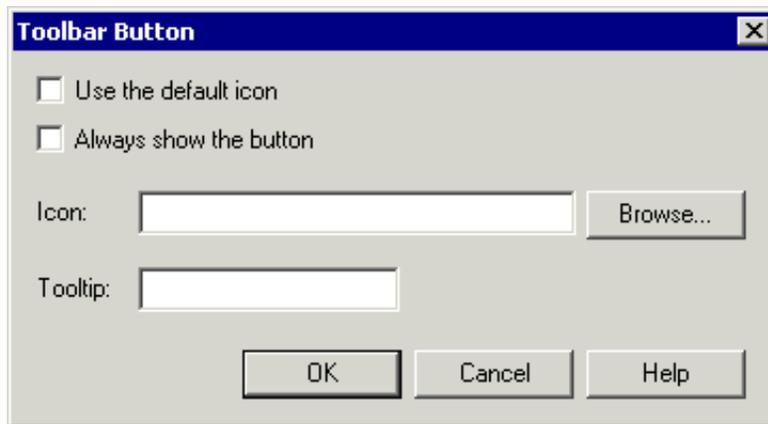
A toolbar button can be any GIF, JPG or BMP file with 16x16 or 19x19 pixels in size.

See *General Information on Image Files* in the documentation *Defining the Rules Using the SDK*.

### ▶ To define an image for a toolbar button

1. Display the Toolbar 3270 Keys dialog box as described above.
2. Select the 3270 key for which you want to define an image.
3. Choose the **Edit** button.

The Toolbar Button dialog box appears.



4. Specify the following information:

#### Use the default icon

Select this check box, if you want to use the default icon that is shipped with Entire Screen Builder.

When this check box is selected, the **Icon** text box and the **Browse** button are not available. However, you can enter a tooltip in the the **Tooltip** text box. If the **Tooltip** text box is empty, the the name of the function that is to be invoked will be used as tooltip.

#### Always show the button

When this check box is selected, the toolbar button is always shown when an icon is to be used (either the default icon or your own image).

When this check box is not selected, only the key that is detected on the character screen is shown. A prerequisite for this is that the location of the key has been defined with the Function Keys rule. For example, when you have defined an image for PA1 and this key is not detected on the character screen, a toolbar button is not shown for this key.

If you define an image and tooltip for a toolbar button, this information overwrites the key name and the name of the function that is to be invoked.

If the check box **Always show the button** is selected, the Function Keys rule is not considered. In this case, toolbar buttons are only shown for the keys for which an image has been defined in the Toolbar Button dialog box.

### Icon

Not available, if **Use the default icon** has been selected.

Specify the path to the image in the rules repository (relative to the root folder of the rules repository) that is to be shown on the toolbar button. You can also choose the **Browse** button to select the file from the Open dialog box.

### Tooltip

Specify a short description that is to appear when the user moves the mouse pointer over this toolbar button.

5. Choose the **OK** button.

# BS2000 Keys Toolbar

Use this basic rule to display a toolbar with buttons for the BS2000 special function keys (F keys, K keys, DUE1 and DUE2 keys).

This chapter covers the following topics:

- Maintaining the BS2000 Keys Toolbar Rule
- Overview of Options
- Defining An Image for a BS2000 Keys Toolbar Button

See also: *BS2000 Keys Toolbar* in the *GUI Viewers* documentation.

---

## Maintaining the BS2000 Keys Toolbar Rule

For each button that is to be shown in the BS2000 keys toolbar, you can either define an image or a default icon.

The default icons are shipped with Entire Screen Builder. They are provided as Windows resources in a DLL file. It is not possible to edit or change the default icons.

### To display all defined toolbar buttons

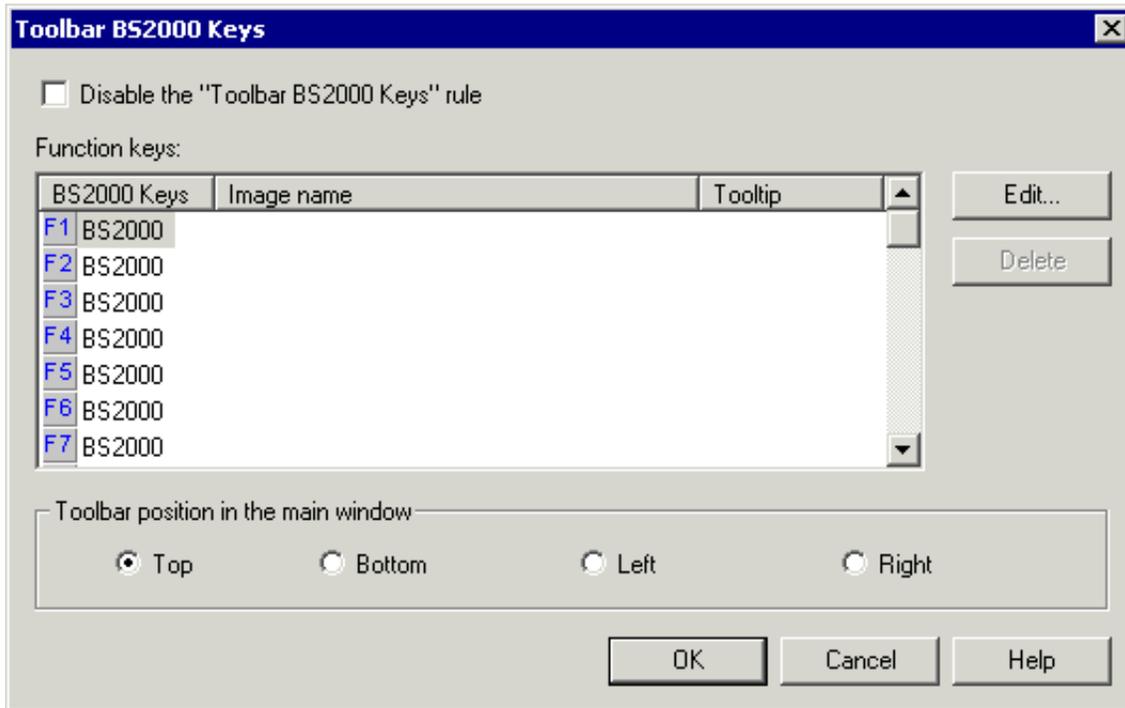
- From the **Basic** menu, choose **Toolbars > BS2000 Keys**.

The Toolbar BS2000 Keys dialog box appears. When images and tooltips have already been defined for a toolbar button, this is shown in this dialog box.

The following command buttons are available:

<b>Edit</b>	Modify the toolbar button definition for the selected BS2000 key. Alternative: double-click a BS2000 key. See <i>Defining An Image for a BS2000 Keys Toolbar Button</i> .
<b>Delete</b>	Delete the toolbar button definition for the selected BS2000 key. The BS2000 key itself is not deleted.

## Overview of Options



### Disable the Toolbar BS2000 Keys rule

When this check box is selected, the BS2000 key toolbar is not shown.

### Toolbar position in the main window

Select an option button to define the position in the window (top, bottom, left or right) at which the toolbar is to be shown.

## Defining An Image for a BS2000 Keys Toolbar Button

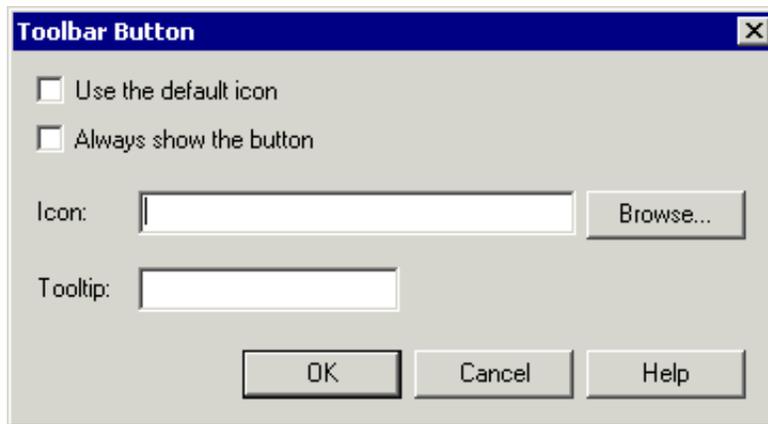
A toolbar button can be any GIF, JPG or BMP file with 16x16 or 19x19 pixels in size.

See *General Information on Image Files* in the documentation *Defining the Rules Using the SDK*.

### ▶ To define an image for a toolbar button

1. Display the Toolbar BS2000 Keys dialog box as described above.
2. Select the BS2000 key for which you want to define an image.
3. Choose the **Edit** button.

The Toolbar Button dialog box appears.



4. Specify the following information:

#### Use the default icon

Select this check box, if you want to use the default icon that is shipped with Entire Screen Builder.

When this check box is selected, the **Icon** text box and the **Browse** button are not available. However, you can enter a tooltip in the the **Tooltip** text box. If the **Tooltip** text box is empty, the the name of the function that is to be invoked will be used as tooltip.

#### Always show the button

When this check box is selected, the toolbar button is always shown when an icon is to be used (either the default icon or your own image).

When this check box is not selected, only the key that is detected on the character screen is shown. A prerequisite for this is that the location of the key has been defined with the Function Keys rule. For example, when you have defined an image for F1 and this key is not detected on the character screen, a toolbar button is not shown for this key.

If you define an image and tooltip for a toolbar button, this information overwrites the key name and the name of the function that is to be invoked.

If the check box **Always show the button** is selected, the Function Keys rule is not considered. In this case, toolbar buttons are only shown for the keys for which an image has been defined in the Toolbar Button dialog box.

### **Icon**

Not available, if **Use the default icon** has been selected.

Specify the path to the image in the rules repository (relative to the root folder of the rules repository) that is to be shown on the toolbar button. You can also choose the **Browse** button to select the file from the Open dialog box.

### **Tooltip**

Specify a short description that is to appear when the user moves the mouse pointer over this toolbar button.

5. Choose the **OK** button.

# Buttons for Main Dialogs

Use this basic rule to create push buttons which correspond to the ENTER key, PF keys, 3270 keys or BS2000 keys for the main dialogs.

This chapter covers the following topics:

- Maintaining the Buttons for Main Dialogs Rule
- Overview of Options
- Defining the Properties for a Push Button

## Maintaining the Buttons for Main Dialogs Rule

The following applies when you have enabled the Function Keys rule and have defined the pattern of the function keys and their location on the characters screens.

You can use the Buttons for Main Dialog rule to create push buttons containing the following:

- the function key label from the character screen (dynamic text),
- an image related to the function key label from the character screen (dynamic image),
- constant text which ignores the function key label from the character screen,
- a constant image which ignores the function key label from the character screen,

### To display all defined push buttons

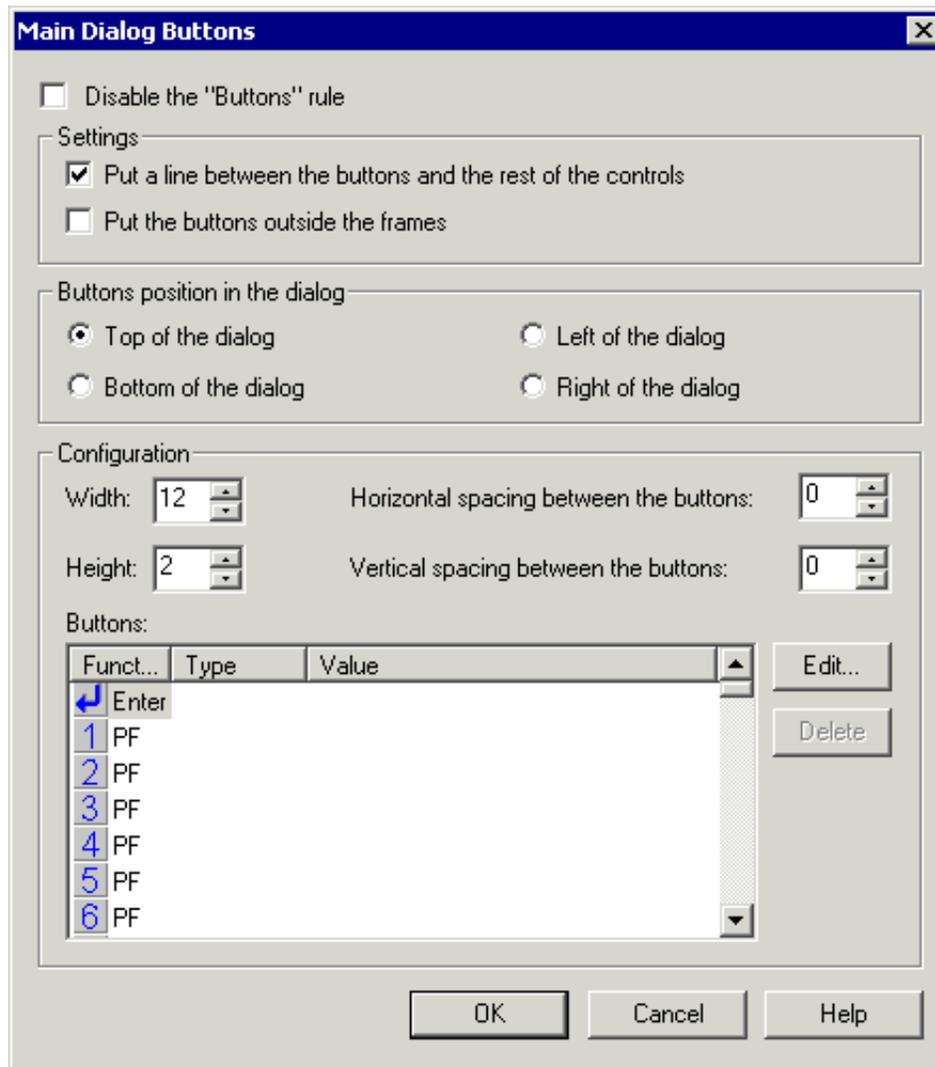
- From the **Basic** menu, choose **Buttons > Main Dialogs**.

The Main Dialog Buttons dialog box appears. When a constant or dynamic image or constant text has already been defined for a push button, this is shown in the Buttons list box.

The following command buttons are available:

<b>Edit</b>	Modify the push button definition for the selected function key. Alternative: double-click a function key. See <i>Defining the Properties for a Push Button</i> .
<b>Delete</b>	Delete the push button definition for the selected function key and reset it to "dynamic text". The function key itself is not deleted.

## Overview of Options



### Disable the Buttons rule

When this check box is selected, this rule is disabled.

### Put a line between the buttons and the rest of the controls

When this check box is selected, a separation line will appear between the push buttons and the dialog.

### Put the buttons outside the frames

If you are using, for example, push buttons at the bottom of the dialog (as defined with this rule) and a frame is also shown at the bottom of the dialog (as defined with the Frames rule), you can select this check box so that the push buttons are moved from the bottom of the dialog to a position below the frame.

If you are using frames and select this check box, the push buttons are moved as indicated in the following table:

<b>Frame name</b>	<b>Position of the push buttons</b>	<b>The buttons are moved to a position ...</b>
"TOP"	Top of the dialog	above the frame.
"BOTTOM"	Bottom of the dialog	below the frame.
"LEFT"	Left of the dialog	to the left of the frame.
"RIGHT"	Right of the dialog	to the right of the frame.

**Note:**

The quotation marks are part of the frame name.

### Buttons position in the dialog

Select an option button to define the position in the dialog (top, bottom, left or right) at which the push buttons are to be shown.

### Configuration

Specify the size (width and height) of the buttons and the distance between the buttons (horizontal and vertical spacing). To do so, specify the number of characters (1-100) in the corresponding spin boxes. For example, a button for which the width and height have been defined with 1, has the same size as a character. When you change the font, the button is automatically resized to the new font size.

## Defining the Properties for a Push Button

The Button Properties dialog box appears when you select a function key in the Main Dialog Buttons dialog box and choose the **Edit** button.

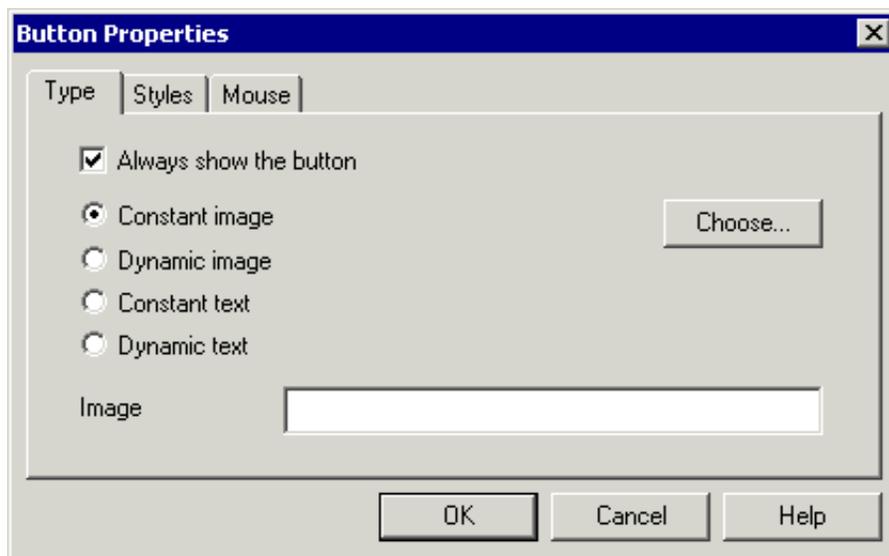
Different pages are available in the Button Properties dialog box:

- Type
- Styles
- Mouse

### Type

The Type page is used to define the content of a push button. A push button corresponds to a specific function key.

The content of a push button can be plain text, a constant image or a dynamic image. See also: *General Information on Image Files* in the documentation *Defining the Rules Using the SDK*.



#### Always show the button

Only available for a constant image or constant text. When this check box is selected, the push button is always shown.

When this check box is not selected, a push button is only shown if the key is detected on the character screen. A prerequisite for this is that the location and pattern of the keys has been defined with the Function Keys rule. For example, when you have defined a constant image for PF1 and this key is not detected on the character screen, a push button is not shown for this key.

### Constant image

A constant image always shows the content of the same file.



The screenshot shows a dialog box with four radio button options: "Constant image" (selected), "Dynamic image", "Constant text", and "Dynamic text". A "Choose..." button is located to the right of the options. Below the options is a text input field labeled "Image".

When you select this option button, you have to specify the path and name of the image (relative to the root folder of the rules repository) in the **Image** text box. You can also choose the **Choose** button to select the image from a list.

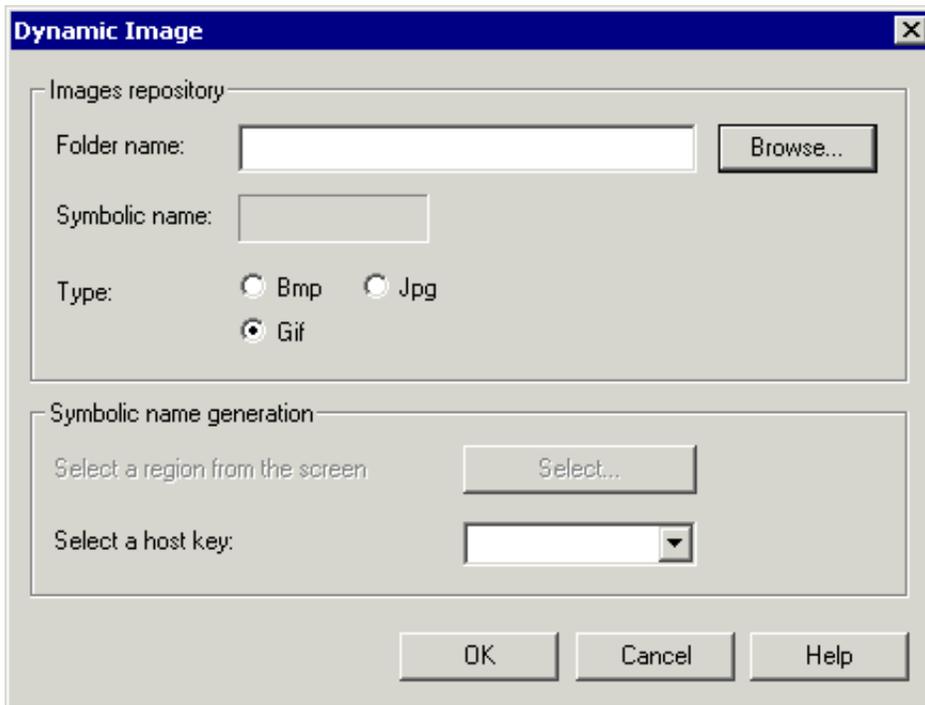
### Dynamic image

A dynamic image shows the content of a file that has the same name as the selected function key.



The screenshot shows a dialog box with four radio button options: "Constant image", "Dynamic image" (selected), "Constant text", and "Dynamic text". A "Choose..." button is located to the right of the options. Below the options is a text input field labeled "Directory".

When you select this option button, you have to choose the **Choose** button. This displays the Dynamic Image dialog box.

**Note:**

The **Select** button is not available for this rule.

Specify the following information:

- **Folder name**  
Specify a folder in your rules repository (relative to the root folder of the rules repository), or choose the **Browse** button to select the folder from a dialog box. This is the folder containing your image files.
- **Type**  
Select the option button for the desired type (**Bmp**, **Gif** or **Jpg**). This is the extension of the image files that are to be used.
- **Select a host key**  
Select the ENTER key, a function key, a 3270 special key or a BS2000 key from this drop-down list box. The name of the selected function key is then shown in the **Symbolic name** text box. It is enclosed in percent (%) signs.

### Constant text

You can define your own text for a push button.



A screenshot of a dialog box for configuring a button. It features four radio button options: 'Constant image', 'Dynamic image', 'Constant text', and 'Dynamic text'. The 'Constant text' option is selected. To the right of these options is a 'Choose...' button. Below the options is a text box labeled 'Caption:'.

When you select this option button, you have to enter the desired text in the **Caption** text box.

### Dynamic text

You can use the text received from the legacy application for a push button.

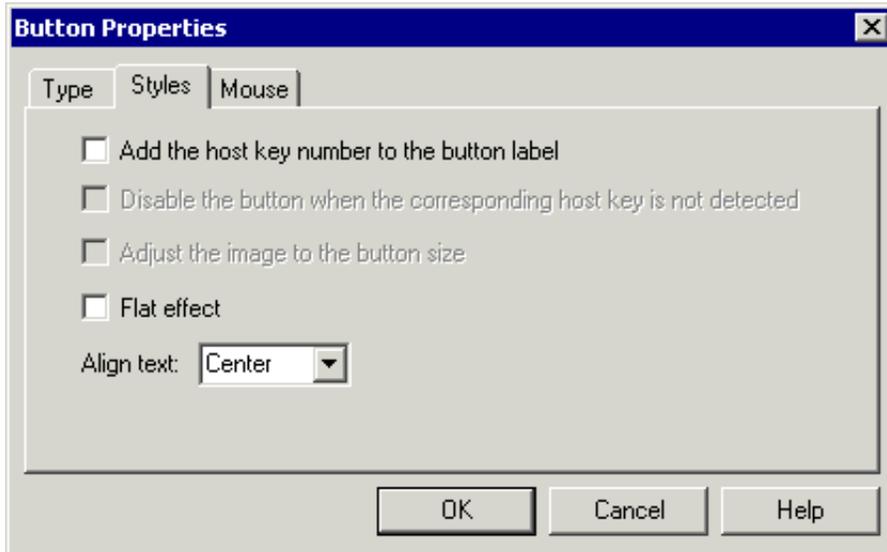


A screenshot of a dialog box for configuring a button. It features four radio button options: 'Constant image', 'Dynamic image', 'Constant text', and 'Dynamic text'. The 'Dynamic text' option is selected. To the right of these options is a 'Choose...' button. Below the options is a text box labeled 'Caption:'.

When you select this option button, no further specifications are required.

## Styles

The Styles page is used to define visual aspects of a push button.



### Add the host key number to the button label

This option is only available if the check box **Always show the button** has not been selected (i.e. only the host keys that are detected on the character screen are shown as push buttons).

When this check box is selected, the name of the corresponding host key is also shown in the label of a push button. For example, "F1-Help". A prerequisite for this is that the content of each push button is defined as dynamic text (see above).

### Disable the button when the corresponding host key is not detected

This option is only available if the check box **Always show the button** has been selected.

When this check box is selected, each push button for which the corresponding host key cannot be found on the screen is disabled.

When this check box is not selected, all push buttons are enabled.

### Adjust the image to the button size

This option is only available for constant and dynamic images.

When this check box is selected, the image is resized so that it fills the whole push button. When this check box is not selected, the image is shown in its original size in the center of the push button.

### Flat effect

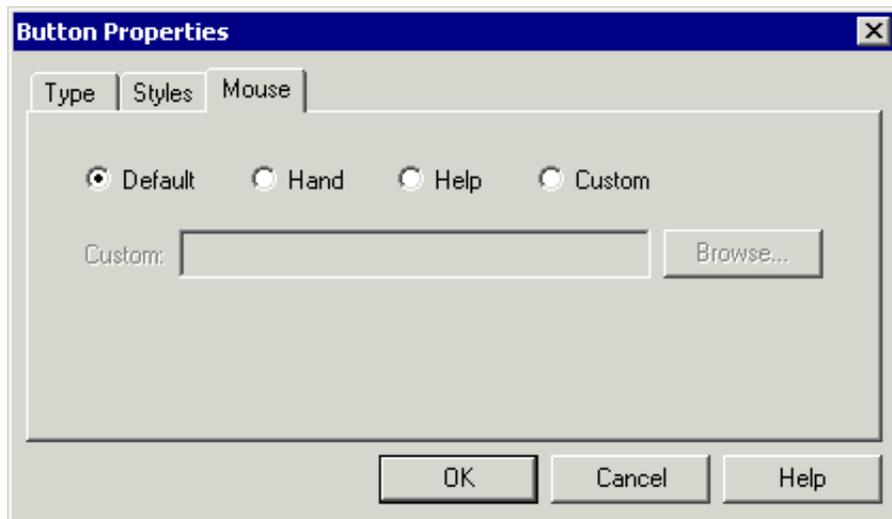
When this check box is selected, the push button will not be shown with a 3D effect.

**Align text**

Specify the alignment of dynamic or constant text on the push button. It can be shown on the left, on the right or in the middle of the push button.

## Mouse

The Mouse page is used to define the appearance of the mouse pointer when it is positioned on a push button.



Select one of the following option buttons:

- **Default**  
The mouse pointer as defined under Windows.
- **Hand**  
Hand-shape mouse pointer.
- **Help**  
Mouse pointer in the shape of a question mark.
- **Custom**  
You can define your own mouse pointer. When this option button has been selected, you must specify a path and file name in the **Custom** text box. This can be a file with the extension *cur* or *ani*. Using the **Browse** button, you can also choose the file from a dialog box.

# Buttons for Child Dialogs

Use this basic rule to create push buttons which correspond to the ENTER key, PF keys, 3270 keys or BS2000 keys for child dialogs.

This chapter covers the following topics:

- Maintaining the Buttons for Child Dialogs Rule
- Overview of Options
- Defining the Properties for a Push Button

## Maintaining the Buttons for Child Dialogs Rule

The following applies when you have enabled the Function Keys rule and have defined the pattern of the function keys and their location on the characters screens.

You can use the Buttons for Child Dialogs rule to create push buttons containing the following:

- the function key label from the character screen (dynamic text),
- an image related to the function key label from the character screen (dynamic image),
- constant text which ignores the function key label from the character screen,
- a constant image which ignores the function key label from the character screen,

### To display all defined push buttons

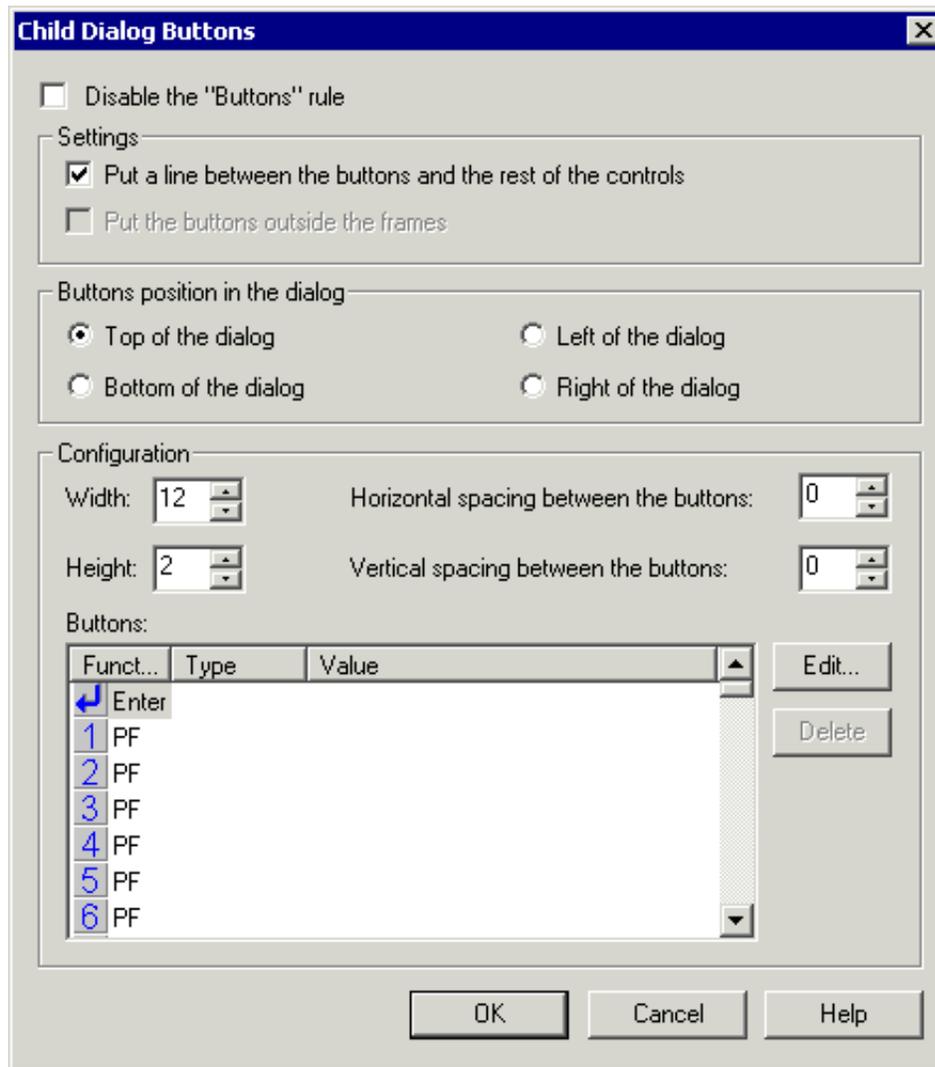
- From the **Basic** menu, choose **Buttons > Child Dialogs**.

The Buttons in the Child Dialogs dialog box appears. When a constant or dynamic image or constant text has already been defined for a push button, this is shown in the Buttons list box.

The following command buttons are available:

<b>Edit</b>	Modify the push button definition for the selected function key. Alternative: double-click a function key. See <i>Defining the Properties for a Push Button</i> .
<b>Delete</b>	Delete the push button definition for the selected function key and reset it to "dynamic text". The function key itself is not deleted.

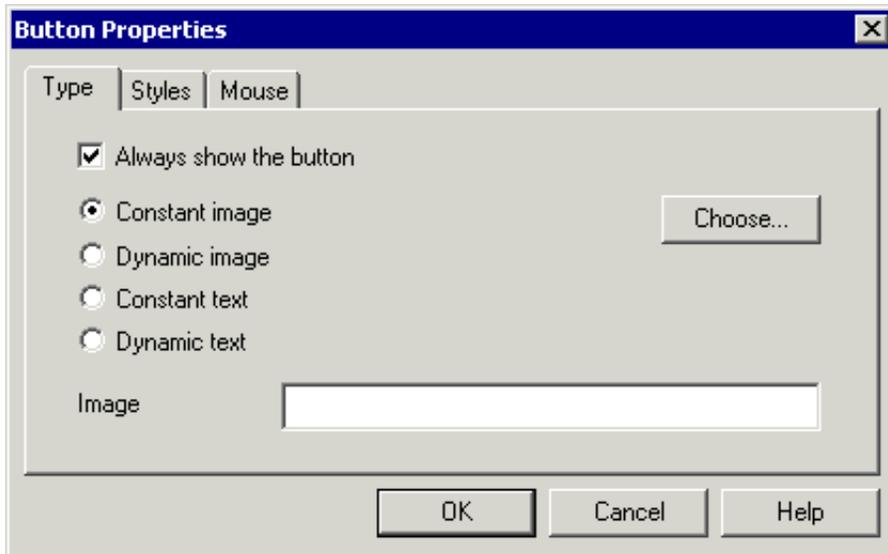
## Overview of Options



This dialog box contains the same information as the dialog box that is shown for the buttons in the main dialogs. The only exception is that buttons cannot be put outside the frames. See the section *Buttons for Main Dialogs* for detailed information on the options in this dialog box.

## Defining the Properties for a Push Button

The Button Properties dialog box appears when you select a function key in the Buttons in the Child Dialogs dialog box and choose the **Edit** button.



See the section *Buttons for Main Dialogs* for detailed information on the options in this dialog box.

# Child Window

Use this basic rule to convert the popup windows in the character screens to child windows.

This chapter covers the following topics:

- Maintaining the Child Window Rules
  - Overview of Options
- 

## Maintaining the Child Window Rules

You have to define a pattern for each popup window that you want to convert to a child window.



**Warning:**

**Do not define the same patterns for the Child Window rule and the Group Box rule.**

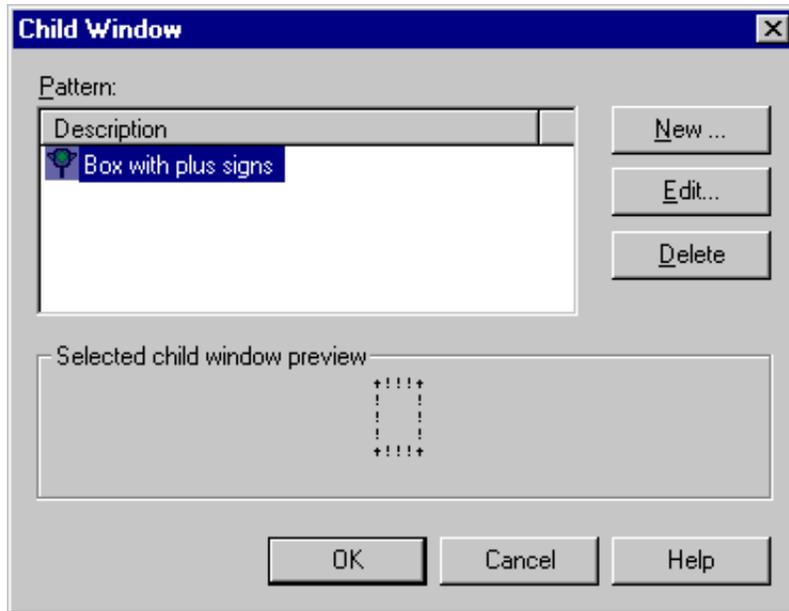
If there is more than one child window on a screen and one of the child windows completely overlaps another child window, the overlapped child window is removed. When the bigger window is moved, the child windows below are no longer displayed.

When the server is a UNIX or OpenVMS machine and Natural statements have been used to create the popup windows, it is not necessary to define this rule. In this case, child windows are detected automatically. However, if the popup windows have been drawn in the Natural map, you have to define this rule so that the popup windows can be converted to child windows.

▶ **To display all defined Child Window rules**

- From the **Basic** menu, choose **Child Window**.

The Child Window dialog box appears.



When patterns have already been defined, the description of each pattern is shown in this dialog box. When you select a description, the defined pattern is shown at the bottom of the dialog box.

The color red in the symbol to the left of a description indicates that this rule has been disabled. A symbol with the color green indicates that the rule is enabled.

The following command buttons are available:

<b>New</b>	Add a new Child Window rule (see below).
<b>Edit</b>	Modify the selected pattern for a Child Window rule. Alternative: double-click a pattern for a Child Window rule.
<b>Delete</b>	Delete the selected Child Window rule.

▶ **To add a Child Window rule**

1. Display the Child Window dialog box as described above.
2. Choose the **New** button.

The Child Window Pattern dialog box appears.

3. Specify all required information as described below.
4. Choose the **OK** button.

## Overview of Options

### Disable this Child Window Pattern

When this check box is selected, this rule is disabled.

### Description

Enter a description for this pattern. A default description is automatically provided.

### Child window preview

Depending on the characters you enter in the following text boxes, this region shows how the popup window in the character screen is to look like.

### Upper left corner

Defines the character that is used in the character screen to indicate the upper left corner of the popup window. You can enter one or more characters in this text box.

### Upper right corner

Defines the character that is used in the character screen to indicate the upper right corner of the popup window. You can enter one or more characters in this text box.

**Vertical char**

Defines the character that is used in the character screen to indicate the vertical lines of the popup window. You can only enter one character in this text box.

**Horizontal char**

Defines the character that is used in the character screen to indicate the horizontal lines of the popup window. You can only enter one character in this text box.

**Down left corner**

Defines the character that is used in the character screen to indicate the lower left corner of the popup window. You can enter one or more characters in this text box.

**Down right corner**

Defines the character that is used in the character screen to indicate the lower right corner of the popup window. You can enter one or more characters in this text box.

**Select**

This command button invokes the Characters Selector dialog box. Instead of typing the characters in the corresponding text boxes, you can also select them from this dialog box. Before choosing a character, position the cursor in the appropriate text box.

**Detect window only if at least one input field is inside the rectangle**

When this check box is selected, a popup window is only converted to a child window if one or more input fields are detected inside the popup window. When this check box is not selected, all rectangles in the screen with the same format as defined in this rule are transformed to child windows.

# Control Attributes

Use this basic rule to define the control size and special characters used in your legacy application.

This chapter covers the following topics:

- Maintaining the Control Attributes Rule
  - Overview of Options
- 

## Maintaining the Control Attributes Rule

When you define this rule, you have to consider the following:

- Check whether the viewer will run when the screen resolution is 640x480.
- If you are working in a UNIX or OpenVMS environment and your Natural application supports field help, you have to know which character is used as the help character.
- In Windows applications, edit boxes do not contain filler characters. If your legacy application uses filler characters, you should remove them using this rule.

### To define the control attributes

1. From the **Basic** menu, choose **Control Attributes**.

The Control Attributes dialog box appears.

2. Specify all required information as described below.
3. Choose the **OK** button.

## Overview of Options

**Control Attributes**

Adjust control's size when its font is changed

Support Standard VGA screens (640x480)

Special input-field characters definition

Help character used in your application:

Filler character used in your application:

Host keys used on the input fields

Host key sent on left button double click:

Host key sent on selection of "Help" :

Host keys used on the output fields

Host key sent on left button double click:

Host key sent on selection of "Help" :

OK Cancel Help

### Adjust control's size when its font is changed

When this check box is selected, the size of a control is automatically increased or reduced according to the font that has been defined for the text in this control.

### Support standard VGA screens (640x480)

It is recommended that you select this check box when VGA screens are used. When this check box is not selected, it may happen that a dialog is larger than can be displayed on the VGA screen.

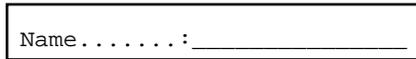
### Help character used in your application

Only applies when your legacy application is running in a UNIX or OpenVMS environment. In this text box, specify the help character that is used in your Natural application. When the user works with the viewer and moves the mouse pointer to a field for which field help has been defined, a question mark is automatically shown with the mouse pointer. When the user then clicks such a field using the right mouse button, the defined help character is sent to the Natural application and field help is displayed.

### Filler character used in your application

In mainframe, UNIX and OpenVMS applications, filler characters are used to differentiate between unprotected and protected fields. Under Windows these filler characters are not required. When you specify the filler character that is used in your legacy application in this text box, the filler characters are no longer shown.

The following example shows a mainframe input field:

A rectangular input field with a thin border. The text "Name . . . . . : \_\_\_\_\_" is displayed inside, where the dots represent filler characters.

When you do not define the filler character, the viewer displays the corresponding control as follows:

A rectangular input field with a thin border. The text "Name : \_\_\_\_\_" is displayed inside, with no filler characters.

However, when you define the filler character, this control looks like a regular Windows edit box:

A rectangular input field with a double-line border, resembling a standard Windows text box. The text "Name : \_\_\_\_\_" is displayed inside.

### Host key sent on left button double-click

This option is available twice. You can define different behavior for input fields and output fields.

Select the host key that is to be sent to the legacy application when the user double-clicks on an input or output field using the left mouse button. By default, the ENTER key is sent.

A BS2000 key can also be sent.

### Host key sent on selection of "Help"

This option is available twice. You can define different behavior for input fields and output fields.

Select the host key that is to be sent to the legacy application when the user clicks the right mouse button on an input or output field and chooses **Help** from the resulting context menu. By default, the PF1 key is sent.

A BS2000 key can be also sent.

# Delete Characters

Use this basic rule to define the characters that are not to be displayed by the viewer.

You can define output fields that are to be deleted and thus not shown in the viewer. If one, and only one, of the characters defined for this rule is contained in an output field, this output field will not be shown. If the output field contains other characters in addition to this character (either defined or undefined), this rule will not be applied. This rule can be used, for example, to delete fields containing the underscore (\_) character. Blanks are ignored.

## Example:

The following characters have been defined in the Delete Characters dialog box:

.\*<>

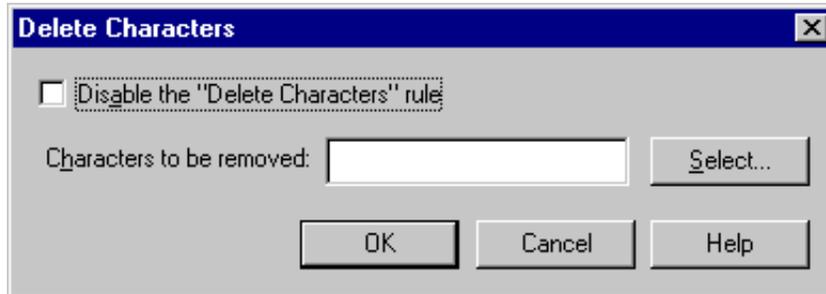
The following table gives examples for fields that may appear in the character screen and informs you whether they are removed by the above definition.

Field in the Character Screen	Remove Characters
<*****>	No
<<<<<<<<<<<	Yes
>>>>>>>>>>	Yes
* * * * * * **	Yes
<** .<<<*** >>>**	No
.....	Yes
U.N.I.C.E.F.	No

▶ To define the characters that are to be deleted

1. From the **Basic** menu, choose **Delete > Characters**.

The Delete Characters dialog box appears. When characters to be deleted have already been defined, they are shown in this dialog box.



2. Either type the characters to be removed in the text box or choose the **Select** button to select them from the Characters Selector dialog box.

Or, if the defined characters are no longer to be removed, select the **Disable the Delete Characters rule** check box.

3. Choose the **OK** button.

# Delete Lines

Use this basic rule to delete lines so that they are not displayed by the viewer.

You can delete any line (1 to 24) of the character screen. For example, you can delete the lines at the top of a screen if you want to place an image with your company logo in this position. You can also delete a line displaying a user ID and then place a dynamic image with a picture of this user in the same position.

This rule shrinks the dialog so that the following lines of the character screen are automatically moved up when displayed with the viewer.

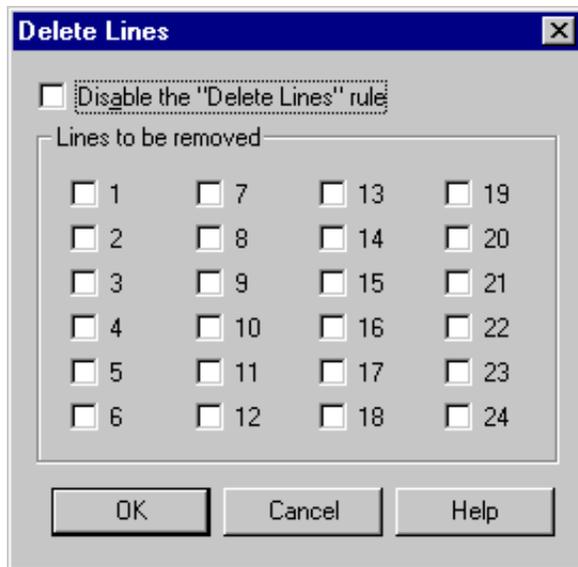
## Important:

If a line contains an input field, it is not deleted.

### ▶ To define the lines that are to be deleted

1. From the **Basic** menu, choose **Delete > Lines**.

The Delete Lines dialog box appears. When lines to be deleted have already been defined, the corresponding check boxes are selected.



2. If you want to remove (additional) lines, select the corresponding check boxes.

Or, if the defined lines are no longer to be removed, select the **Disable the Delete Lines rule** check box.

3. Choose the **OK** button.

# Delete Prompt

Use this basic rule to define the prompts that are not to be displayed.

This chapter covers the following topics:

- Maintaining the Delete Prompt Rules
- Overview of Options

## Maintaining the Delete Prompt Rules

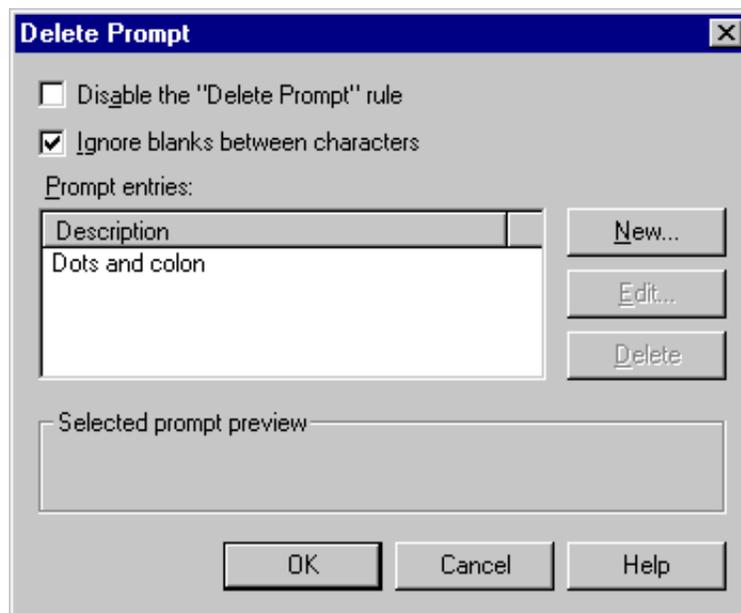
This rule is useful, if you want to remove the prompt characters, consisting of filler and ending characters, from the label of a field. In the following example, "." is the filler character and ":" is the ending character:

NAME .....:

### ▶ To display all defined Delete Prompt rules

1. From the **Basic** menu, choose **Delete > Prompt**.

The Delete Prompt dialog box appears.



When prompt entries have already been defined, the description of each entry is shown in this dialog box. When you select a description, the defined prompt is shown at the bottom of the dialog box.

2. If you want to disable all defined Delete Prompt rules, select the **Disable the Delete Prompt rule** check box.

Or:

If you want to define a new Delete Prompt rule, proceed as described below.

3. Select the **Ignore blanks between characters** check box, if the defined prompts are to be deleted even if they contain blanks. For example:

Name . . . . :
----------------

The following command buttons are available:

<b>New</b>	Add a new Delete Prompt rule (see below).
<b>Edit</b>	Modify the selected pattern for a Delete Prompt rule. Alternative: double-click a pattern for a Delete Prompt rule.
<b>Delete</b>	Delete the selected Delete Prompt rule.

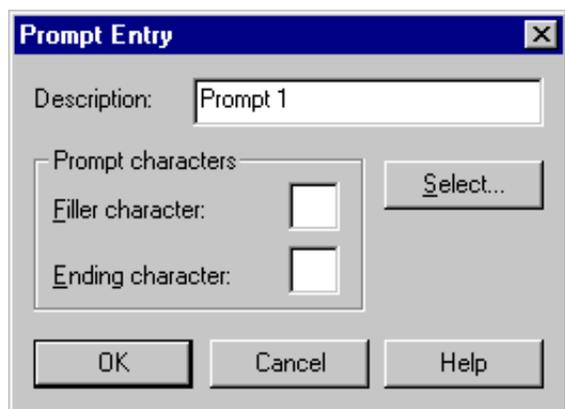
 **To add a Delete Prompt rule**

1. Display the Delete Prompt dialog box as described above.
2. Choose the **New** button.

The Prompt Entry dialog box appears.

3. Specify all required information as described below.
4. Choose the **OK** button.

## Overview of Options



### Description

Enter a description for this prompt. A default description is automatically provided.

### Filler character

Defines the character that is used as the filler character in the character screen.

### Ending character

Defines the character that is used as the ending character in the character screen.

### Select

This command button invokes the Characters Selector dialog box. Instead of typing the characters in the corresponding text boxes, you can also select them from this dialog box. Before choosing a character, position the cursor in the appropriate text box.

# Font and Colors for Main Dialogs

Use this basic rule to define the background color for the main dialogs, and the font, style, size and/or color for the following attributes: normal, underline, intensified, blinking and reverse.

You can also define that the colors are used as sent from host. The following applies for the different operating systems:

- **Mainframe**

Foreground and background colors are taken from the color scheme that has been defined using the System Management Hub. See *Color Schemes* in Entire Screen Builder's *System Management Hub* documentation.

Blinking is not supported. This attribute is ignored.

- **UNIX and OpenVMS**

The foreground color is taken from Natural.

The background color is taken from the Font and Colors dialog box (see below).

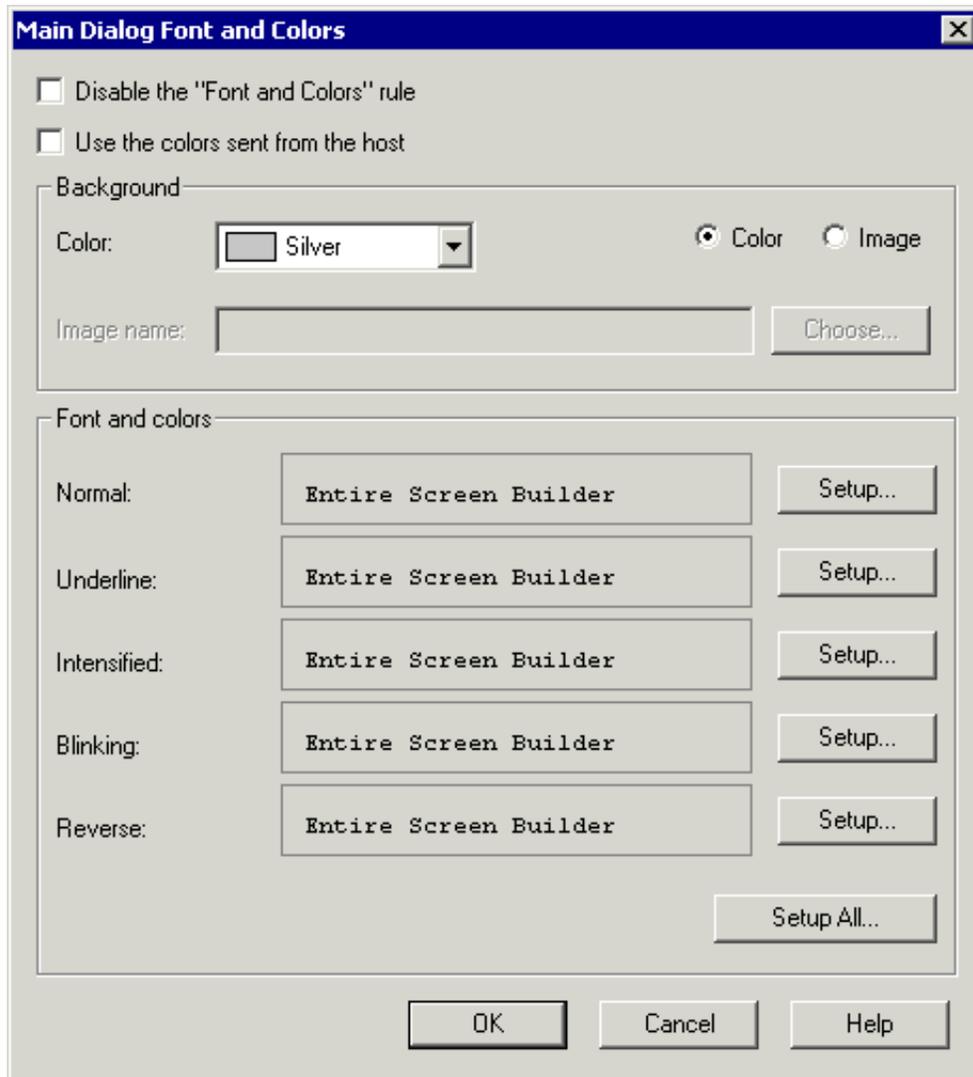
The font is always taken from the **Normal** option in the Font and Colors dialog box (see below).

Instead of a background color, you can also define a background image. In this case, all controls generated using the basic rules are transparent. Static text controls and edit box controls that result from the built-in rules are also transparent. Exception: if you are using the basic rule Frames, you must define the "Transparent" style for the dialogs using the resource editor.

▶ To modify font and/or color for the main dialogs

1. From the **Basic** menu, choose **Font and Colors > Main Dialogs**.

The Main Dialog Font and Colors dialog box appears showing the currently defined fonts and colors for the main dialogs.



2. If you want to disable the current settings, select the **Disable the Font and Colors rule** check box.

Or:

If you want to change the font and/or color, proceed as described below.

3. If you want to use the colors that are sent from the host, select the corresponding check box. In this case, the only option that is available in the Font and Colors group box is **Normal**.
4. If you want to define the background color for the dialogs, make sure that the **Color** option button is selected and select the desired color from the **Color** drop-down list box. When you select **System** from the drop-down list box, the background color as defined under Windows will be used. When you select **Custom...** from the drop-down list box, the Color dialog box appears and you can select additional colors or define custom colors. The currently defined color is shown next to **Custom...**

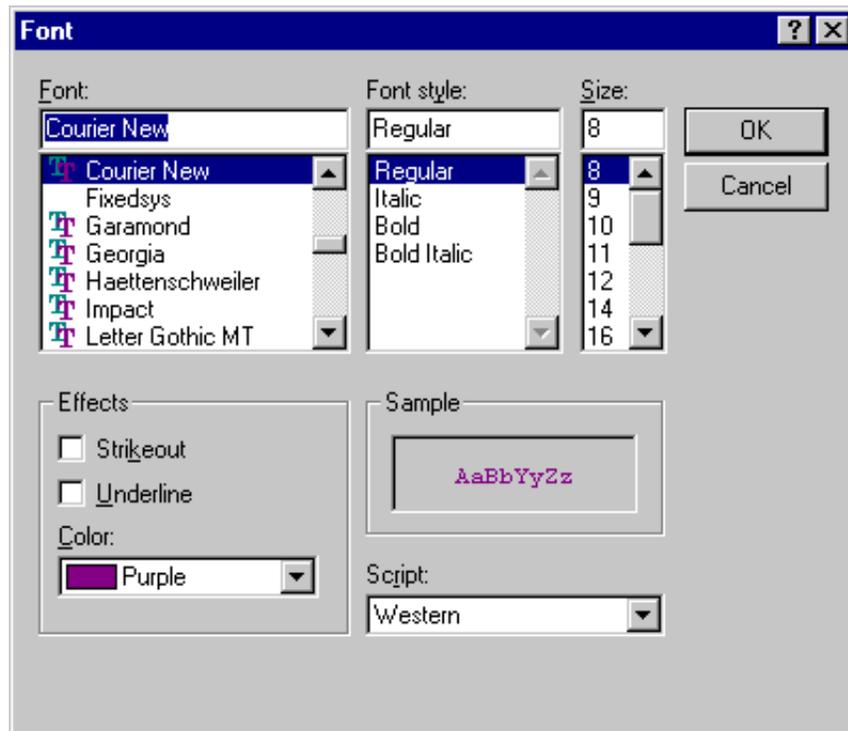
Or:

If you want to use a background image instead of a background color, make sure that the **Image** option button is selected. In the **Image name** text box, specify the path to an image in the rules repository (relative to the root folder of the rules repository). You can also choose the **Choose** button to select the file from the "Open" dialog box. The background image can be any GIF, JPG or BMP file. See *General Information on Image Files* in the documentation *Defining the Rules Using the SDK*.

5. If you want to define different fonts and/or colors for the different attributes, choose the corresponding **Setup** button for an attribute.

Or, if you want to define the same font and/or color for all attributes, choose the **Setup All** button.

In both cases, the Font dialog box appears.



6. Specify all required options and choose the **OK** button.

The Main Dialog Font and Colors dialog box now shows the new settings for the attributes.

7. Choose the **OK** button to close the Main Dialog Font and Colors dialog box.

# Font and Colors for Child Dialogs

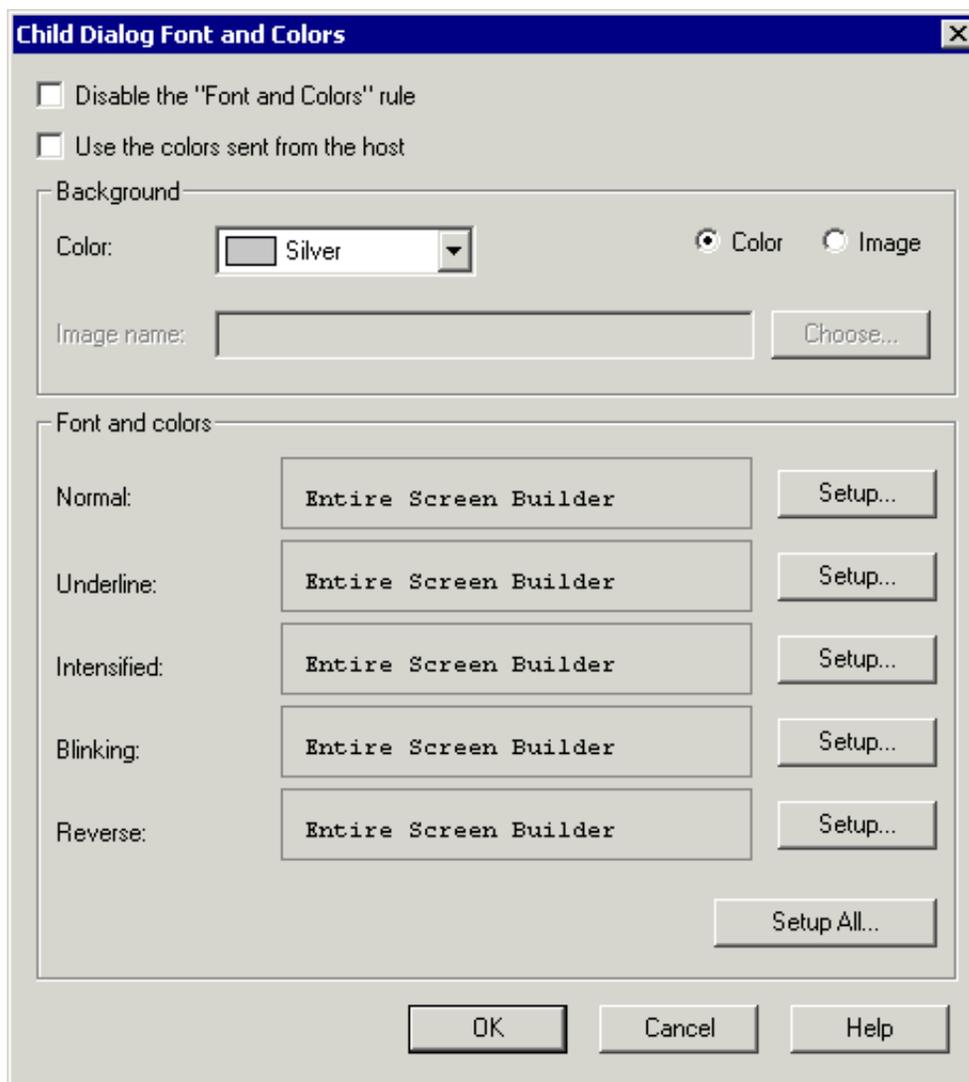
Use this basic rule to define the background color for the child dialogs, and the font, style, size and/or color for the following attributes: normal, underline, intensified, blinking and reverse.

The information given for the main dialogs also applies for the child dialogs. See *Font and Colors for Main Dialogs* for detailed information.

## ▶ To modify font and/or color for the child dialogs

1. From the **Basic** menu, choose **Font and Colors > Child Dialogs**.

The Child Dialog Font and Colors dialog box appears showing the currently defined fonts and colors for the child dialogs.



This dialog box contains the same information as the dialog box that is shown for the font and colors in the main dialogs.

2. Proceed as described in the section *Font and Colors for Main Dialogs*.

# Frames

Use this basic rule to define up to four dialogs for which extended rules have been defined. These dialogs are used as frames.

The frames can be placed at the top, bottom, left and right of the dialog containing the basic rules. This is helpful, for example, if you always want to place the same image at the top of a dialog and the same buttons for the PF keys at the bottom of a dialog. The end-user cannot see that different frames are used.

The dialogs to be used as frames must be contained in a BDD file. The dialogs must be named as follows (the quotation marks are part of the name):

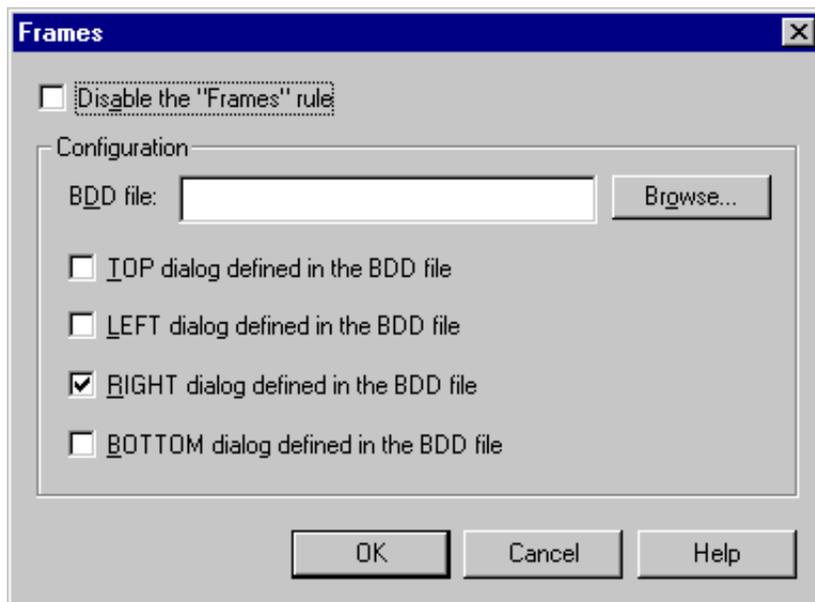
- "TOP"
- "LEFT"
- "RIGHT"
- "BOTTOM"

See *Building the BDD File* in the documentation *Defining the Rules Using the SDK* for further information.

## ▶ To define the Frames rule

1. From the **Basic** menu, choose **Frames**.

The Frames dialog box appears. When frames have already been defined, this information is shown in this dialog box.



2. If the defined frames are no longer to be shown, select the **Disable the Frames rule** check box.

Or:

Modify the information in this text box as described below.

3. In the **BDD file** text box, specify the path to the BDD file.

Or:

Choose the **Browse** button to select the BDD file from the Open dialog box.

4. Select the corresponding check box for each dialog that is to be used as a frame.
5. Choose the **OK** button.

# Function Keys

Use this basic rule to define the detection logic for the function keys in the character screens. To do so, you have to define the lines that are to be searched for function keys and an identifier for each function key that is to be detected.

The following keys can be detected:

- the ENTER key,
- function keys (PF1 through PF48),
- 3270 special function keys (PA1, PA2, PA3, Att, Reset, Clear or SysReq),
- BS2000 special function keys (F1 through F24, K1 through K14, DUE1 and DUE2).

The following applies when the host is a UNIX or OpenVMS machine and when the function keys have been defined using Natural statements: if the environment variable `NSW_PF_MSG_LINES_NATIVE_FORMAT` has not been set to "YES", it is not necessary to define this rule. In this case, the function keys are detected automatically by Natural UNIX and Natural OpenVMS. See the sections *Installing Entire Screen Builder on Natural UNIX Hosts* and *Installing Entire Screen Builder on Natural OpenVMS Hosts* in the *Installation and Configuration* documentation for further information on this environment variable. However, if the function keys have been defined manually in the Natural map, you must also define the function keys using this rule.

This chapter covers the following topics:

- Maintaining the Function Keys Rules
  - Overview of Options
  - Defining the Detection Rule for a Specific Function Key
-

## Maintaining the Function Keys Rules

When function keys are detected on a screen, they are removed from the character screen. They are then added to one of the following toolbars - depending on the session type:

- function keys toolbar
- 3270 keys toolbar
- BS2000 keys toolbar

To send PF keys in AS/400 style, the session has to be defined as an AS/400 session. See *Communication Properties for Telnet TN3270* in Entire Screen Builder's *System Management Hub* documentation.

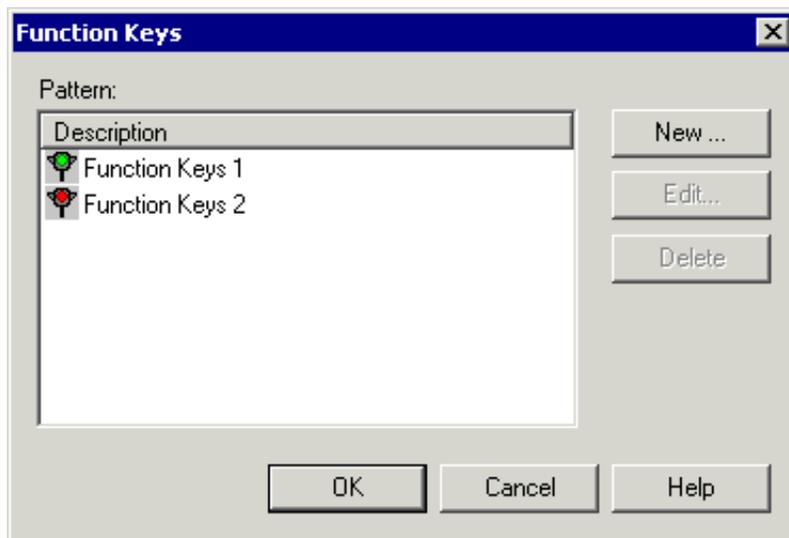
If you do not define a Function Keys rule, function keys cannot be detected by the following rules:

- Buttons for Main Dialogs
- Buttons for Child Dialogs

### ▶ To display all defined Function Keys rules

- From the **Basic** menu, choose **Function Keys**.

The Function Keys dialog box appears.



When function key rules have already been defined, the description of each rule is shown in this dialog box.

The color red in the symbol to the left of a description indicates that this rule has been disabled. A symbol with the color green indicates that the rule is enabled.

#### **Caution:**

If the same key has been defined in two different rules which are both active, the definition from the first rule in the above dialog box is used.

The following command buttons are available:

<b>New</b>	Add a new Function Keys rule (see below).
<b>Edit</b>	Modify the selected Function Keys rule. Alternative: double-click a a Function Keys rule.
<b>Delete</b>	Delete the selected Function Keys rule.

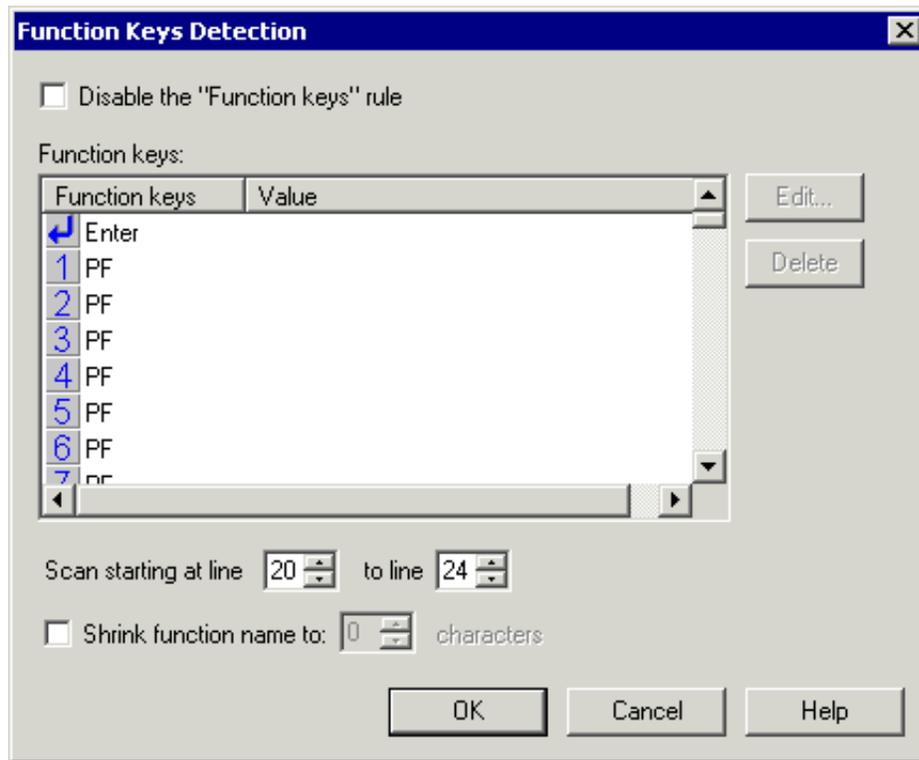
 **To add a Function Keys rule**

1. Display the Function Keys dialog box as described above.
2. Choose the **New** button.

The Function Keys Detection dialog box appears.

3. Specify all required information as described below.
4. Choose the **OK** button.

## Overview of Options



When a detection rule has already been defined for a specific key, this is shown in the Value column.

### Disable the Function Keys rule

When this check box is selected, this rule is disabled.

### Scan starting at line *n* to line *n*

The lines of the screen that are to be searched for the function keys. Example: lines 20 through 24.

### Shrink function name to *n* characters

When this check box is selected, the names of the functions that are to be invoked are truncated to the number of characters specified in the spin box.

The following command buttons are available:

<b>Edit</b>	Add/modify the detection rule for the selected function key (see below). Alternative: double-click a function key.
<b>Delete</b>	Delete the detection rule for the selected function key. The function key itself is not deleted.

## Defining the Detection Rule for a Specific Function Key

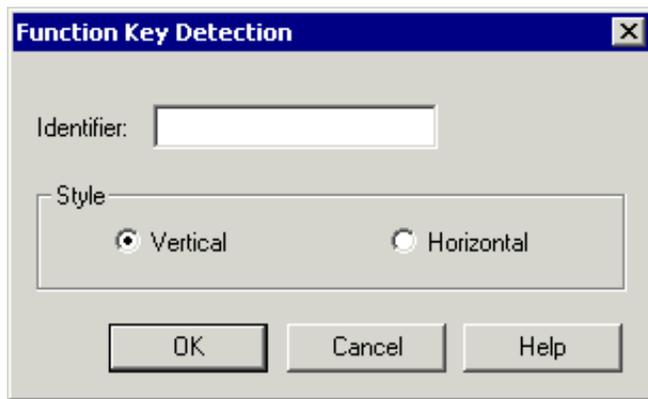
You have to define a detection rule (identifier), if you want to detect the following:

- a function key on a mainframe screen,
- a function key that has been added manually to a Natural UNIX or Natural OpenVMS screen.

### ▶ To define the detection rule for a specific function key

1. Display the Function Keys Detection dialog box as described above.
2. Select the function key for which you want to define a detection rule.
3. Choose the **Edit** button.

The Function Key Detection dialog box appears.



4. In the **Identifier** text box, specify a case-sensitive string that is considered as an indicator of a possible function key. The identifier depends on the style (see below).
5. In the Style group box, select one of the following option buttons:

- **Vertical**

The key name is above the function name that is invoked with this key.

Example:

Enter	PF1	PF2	PF3
Intro	Help	Exit	

If you want to define the detection rule for PF1, you have to define "PF1" in the **Identifier** text box.

- **Horizontal**

The key name is to the left of the function name that is invoked with this key.

Example:

```
Enter=Intro PF1=Help PF3=Exit
```

If you want to define the detection rule for PF1, you have to define "PF1=" in the **Identifier** text box.

6. Choose the **OK** button.

# Function Keys Toolbar

Use this basic rule to display a toolbar with buttons for function keys (PF keys and ENTER key).

This chapter covers the following topics:

- Maintaining the Function Keys Toolbar Rule
- Overview of Options
- Defining An Image for a Function Keys Toolbar Button

See also: *Function Keys Toolbar* in the *GUI Viewers* documentation.

---

## Maintaining the Function Keys Toolbar Rule

For each button that is to be shown in the function keys toolbar, you can either define your own image or a default icon.

The default icons are shipped with Entire Screen Builder. They are provided as Windows resources in a DLL file. It is not possible to edit or change the default icons.

### To display all defined toolbar buttons

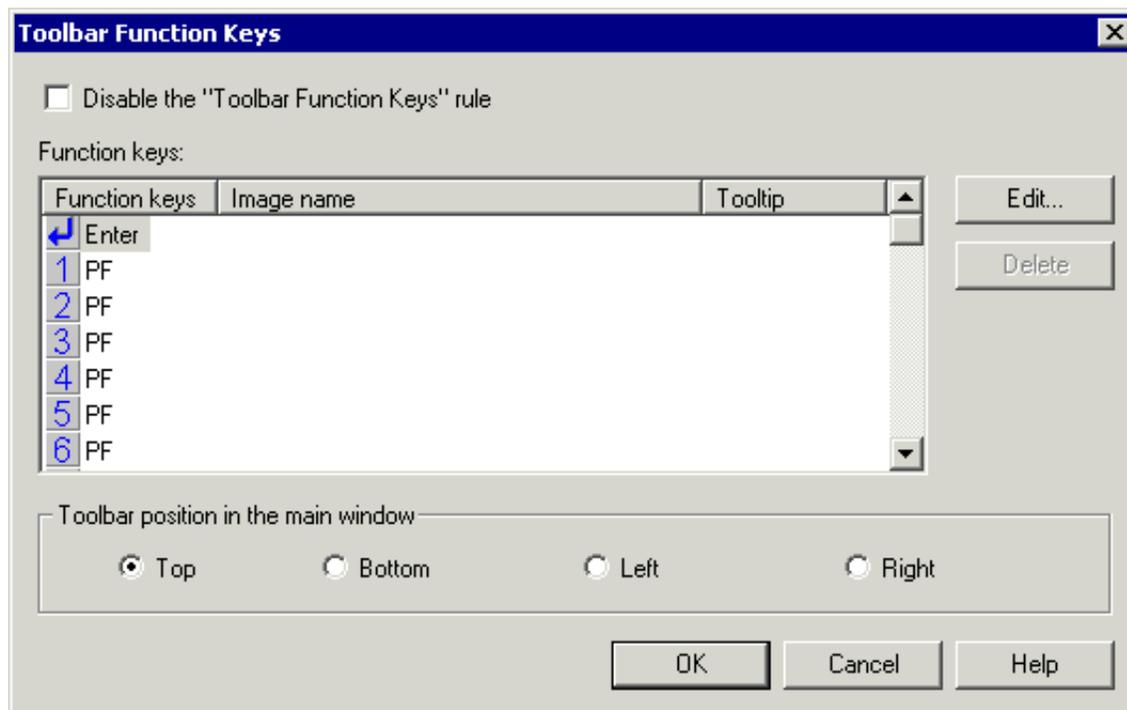
- From the **Basic** menu, choose **Toolbars > Function Keys**.

The Toolbar Function Keys dialog box appears. When images and tooltips have already been defined for a toolbar button, this is shown in this dialog box.

The following command buttons are available:

<b>Edit</b>	Modify the toolbar button definition for the selected function key. Alternative: double-click a function key. See <i>Defining An Image for a Function Keys Toolbar Button</i> .
<b>Delete</b>	Delete the toolbar button definition for the selected function key. The function key itself is not deleted.

## Overview of Options



### Disable the Toolbar Function Keys rule

When this check box is selected, the function key toolbar is not shown.

### Toolbar position in the main window

Select an option button to define the position in the window (top, bottom, left or right) at which the toolbar is to be shown.

## Defining An Image for a Function Keys Toolbar Button

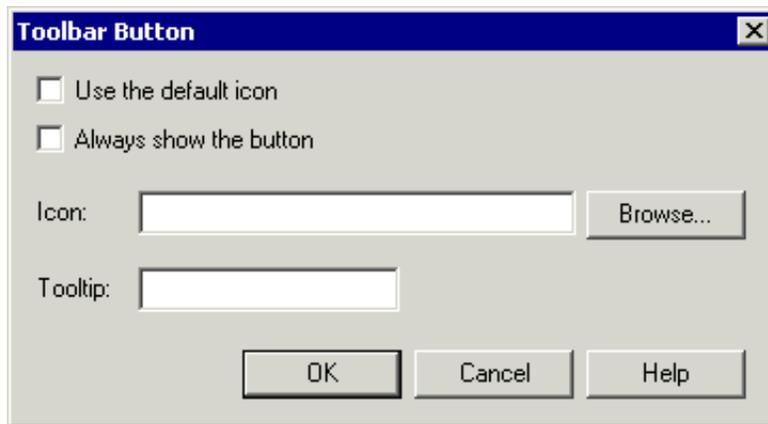
A toolbar button can be any GIF, JPG or BMP file with 16x16 or 19x19 pixels in size.

See *General Information on Image Files* in the documentation *Defining the Rules Using the SDK*.

### ▶ To define an image for a toolbar button

1. Display the Toolbar Function Keys dialog box as described above.
2. Select the function key for which you want to define an image.
3. Choose the **Edit** button.

The Toolbar Button dialog box appears.



4. Specify the following information:

#### Use the default icon

Select this check box, if you want to use the default icon that is shipped with Entire Screen Builder.

When this check box is selected, the **Icon** text box and the **Browse** button are not available. However, you can enter a tooltip in the the **Tooltip** text box. If the **Tooltip** text box is empty, the name of the function that is to be invoked will be used as tooltip.

#### Always show the button

When this check box is selected, the toolbar button is always shown when an icon is to be used (either the default icon or your own image).

When this check box is not selected, only the key that is detected on the character screen is shown. A prerequisite for this is that the location of the key has been defined with the Function Keys rule. For example, when you have defined an image for PF1 and this key is not detected on the character screen, a toolbar button is not shown for this key.

If you define an image and tooltip for a toolbar button, this information overwrites the key name and the name of the function that is to be invoked.

If the check box **Always show the button** is selected, the Function Keys rule is not considered. In this case, toolbar buttons are only shown for the keys for which an image has been defined in the Toolbar Button dialog box.

### Icon

Not available, if **Use the default icon** has been selected.

Specify the path to the image in the rules repository (relative to the root folder of the rules repository) that is to be shown on the toolbar button. You can also choose the **Browse** button to select the file from the Open dialog box.

### Tooltip

Specify a short description that is to appear when the user moves the mouse pointer over this toolbar button.

5. Choose the **OK** button.

# Group Box

Use this basic rule to convert boxes in the character screens to group boxes.

This chapter covers the following topics:

- Maintaining the Group Box Rules
  - Overview of Options
- 

## Maintaining the Group Box Rules

You have to define a pattern for each box that you want to convert to a group box.



**Warning:**

**Do not define the same patterns for the Group Box rule and the Child Window rule.**

Two types of pattern are available:

- One for detecting all characters that are used to make up the box.
- Another for using diagonal detection (the upper left and the lower right corners). If you want to place a title in the group box, you have to define the title mark character. With diagonal detection you have to modify the map of your legacy application: insert an output field in the upper left corner and another output field in the lower right corner. Set the Natural attribute "no display" for both fields. It is important that you use characters that are not used in your legacy application.

**Example for Diagonal Detection:**

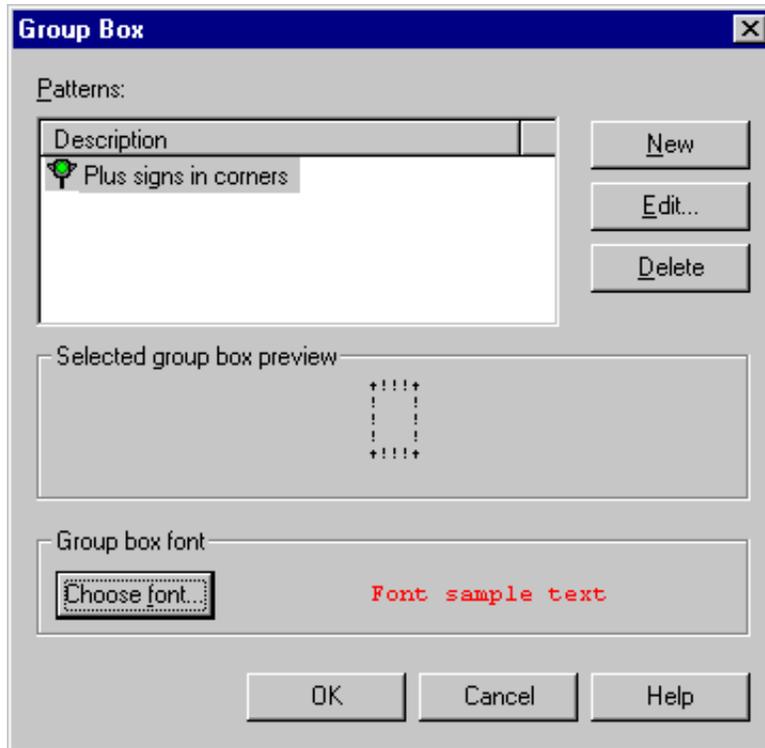
In the following example, the character "{" is used for the upper left corner and "}" is used for the lower right corner. The hash character (#) is used as the title mark.

```
SOFTWARE AG      *** EMPLOYEE MANAGEMENT ***      21/04/99
NSWEMPAL        ADD EMPLOYEES                      17:45:09
-----
{ #PERSONAL DATA
CODE.....:_____
NAME.....:_____ SURNAME.....:_____
MARITAL ST:___ DATE OF BIRTH..:___ ___ ___
}
{ #ADDRESS
ADDRESS...:_____ CITY.....:_____
COUNTRY...:___ POST CODE.:_____
}
{ #COMPANY DATA
DEPARTMENT.....:_____
CATEGORY.....:_____
}
```

▶ To display all defined Group Box rules

- From the **Basic** menu, choose **Group Box**.

The Group Box dialog box appears.



When patterns have already been defined, the description of each pattern is shown in this dialog box. When you select a description, the defined pattern is shown as a preview.

The color red in the symbol to the left of a description indicates that this rule has been disabled. A symbol with the color green indicates that the rule is enabled.

The following command buttons are available:

<b>New</b>	Add a new Group Box rule (see below).
<b>Edit</b>	Modify the selected pattern for a Group Box rule. Alternative: double-click a pattern for a Group Box rule.
<b>Delete</b>	Delete the selected Group Box rule.
<b>Choose font</b>	Invoke the Font dialog box in which you can define the font and/or color for the title that is shown in the frame of the group box. The currently defined font and color, which applies to all defined patterns, is shown next to this command button.

 **To add a Group Box rule**

1. Display the Group Box dialog box as described above.
2. Choose the **New** button.

The Group Box Pattern dialog box appears.

3. Specify all required information as described below.
4. Choose the **OK** button.

## Overview of Options

### Disable this Group Box rule

When this check box is selected, this rule is disabled.

### Description

Enter a description for this pattern. A default description is automatically provided.

### Group box preview

Depending on the characters you enter in the following text boxes, this region shows how the box in the character screen is to look like.

### Upper left corner

Defines the character that is used in the character screen to indicate the upper left corner of the box. You can enter one or more characters in this text box.

### Upper right corner

Not available when the **Use diagonal detection** check box is selected.

Defines the character that is used in the character screen to indicate the upper right corner of the box. You can enter one or more characters in this text box.

**Vertical char**

Not available when the **Use diagonal detection** check box is selected.

Defines the character that is used in the character screen to indicate the vertical lines of the box. You can only enter one character in this text box.

**Title mark char**

Only available when the **Use diagonal detection** check box is selected.

Defines the character that is used in the character screen to detect the group box title. You can only enter one character in this text box.

**Horizontal char**

Not available when the **Use diagonal detection** check box is selected.

Defines the character that is used in the character screen to indicate the horizontal lines of the box. You can only enter one character in this text box.

**Down left corner**

Not available when the **Use diagonal detection** check box is selected.

Defines the character that is used in the character screen to indicate the lower left corner of the box. You can enter one or more characters in this text box.

**Down right corner**

Defines the character that is used in the character screen to indicate the lower right corner of the box. You can enter one or more characters in this text box.

**Use diagonal detection (only upper-left and down-right corners)**

When this check box is selected, you can define the title mark character. In this case, it is only possible to define the upper left and down right corners.

**Select**

This command button invokes the Characters Selector dialog box. Instead of typing the characters in the corresponding text boxes, you can also select them from this dialog box. Before choosing a character, position the cursor in the appropriate text box.

# Image

Use this basic rule to display images in the dialog.

This chapter covers the following topics:

- Maintaining the Image Rules
  - Overview of Options
  - Defining Constant and Dynamic Images
- 

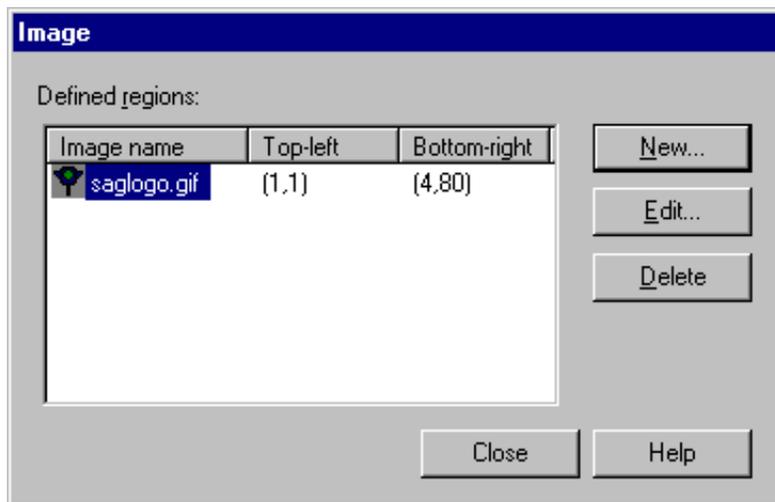
## Maintaining the Image Rules

The images can be stored in any folder of the rules repository. See *General Information on Image Files* in the documentation *Defining the Rules Using the SDK*.

### ▶ To display all defined Image rules

- From the **Basic** menu, choose **Image**.

The Image dialog box appears. When this dialog box is shown, a check mark is shown next to the **Image** command.



When images have already been defined, the defined regions are shown in this dialog box. The scope window then shows the outlines of all defined images.

The color red in the symbol to the left of each defined region indicates that this rule has been disabled. A symbol with the color green indicates that the rule is enabled.

The following command buttons are available:

<b>New</b>	Add a new Image rule (see below).
<b>Edit</b>	Modify the selected Image rule. Alternative: double-click an Image rule.
<b>Delete</b>	Delete the selected Image rule. Alternative: choose <b>Delete</b> from the <b>Edit</b> menu or press DEL to delete the selected Image rule.

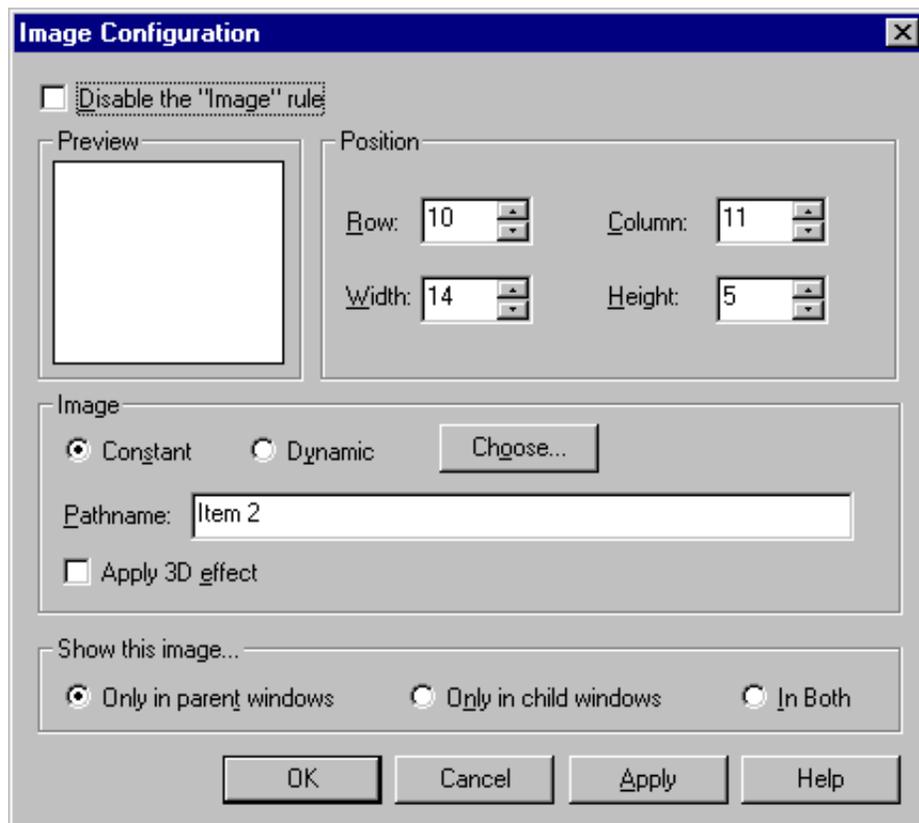
▶ **To add an Image rule**

1. Make sure that the screen file that is to be used as a template is shown in the scope window.
2. Display the Image dialog box as described above.
3. Choose the **New** button.
4. In the scope window, use the mouse to select the region that is to contain the image.

The Image Configuration dialog box appears.

5. Specify all required information as described below.
6. Choose the **OK** button.

## Overview of Options



### Disable the Image rule

When this check box is selected, this Image rule is disabled.

### Preview

Only applies to constant images. Shows the image defined in the **Pathname** text box.

### Position

The dialog position (row, column, width and height) at which the image is to be inserted. This is the region that has been selected using the mouse. Using the spin buttons, you can manually adjust the values.

### Constant / Dynamic / Pathname

Select one of the following option buttons: **Constant** or **Dynamic**. See the description below for further information.

### Apply 3D effect

When this check box is selected, the image is shown with a 3D effect.

### Show this image

Select an option button to define whether the image is to be shown

- only in parent windows,
- only in child windows, or
- in both parent and child windows.

## Defining Constant and Dynamic Images

A constant image always shows the content of the same file.

A dynamic image shows the content of a file that has the same name as the string that is found in the defined region of the screen. For example, when the content of this field is a user ID, an image that has the same name as this user ID will be displayed.

### ▶ To define a constant image

1. Select the **Constant** option button.
2. Choose the **Choose** button.

The Open dialog box appears.

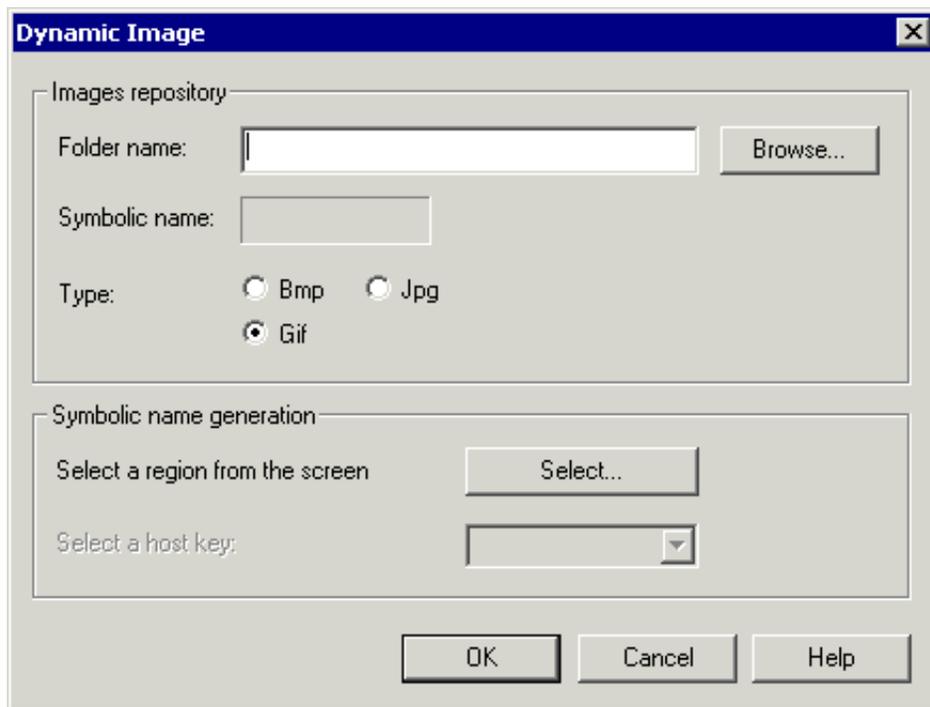
3. Select the desired image.
4. Choose the **Open** button.

The path is now shown in the **Pathname** text box.

### ▶ To define a dynamic image

1. Select the **Dynamic** option button.
2. Choose the **Choose** button.

The Dynamic Image dialog box appears.



3. Specify a folder in your rules repository (relative to the root folder of the rules repository), or choose the **Browse** button to select the folder from a dialog box.

This is the folder containing your image files.

4. Select the option button for the desired type (**Bmp**, **Gif** or **Jpg**).

This is the extension of the image files that are to be used.

5. Choose the **Select** button.
6. Select a region in the scope window.

The content of this region is then used to display an image with the same name as the character found in the defined region. The location of the defined region is now shown in the **Symbolic name** text box. It is enclosed in percent (%) signs.

**Note:**

The **Select a host key** drop-down list box is not available for this rule.

7. Choose the **OK** button.

The path is now shown in the Image Configuration dialog box.

# Item

Use this basic rule to move or copy text from the character screen to the title bar or status bar.

This chapter covers the following topics:

- Maintaining the Item Rules
- Overview of Options

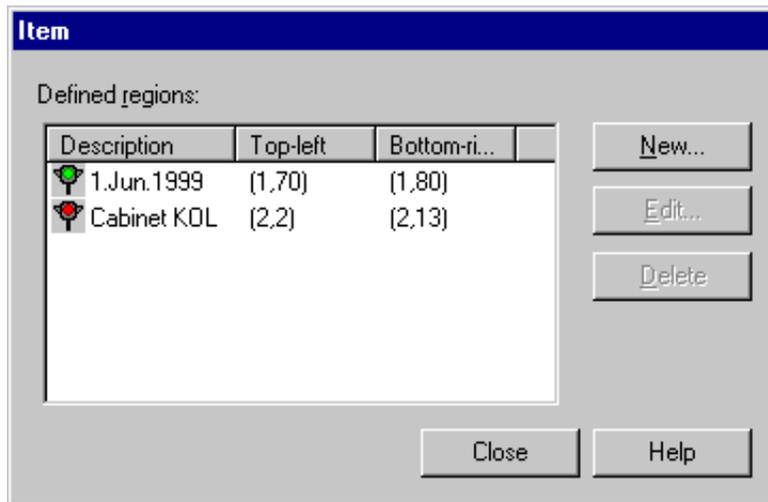
## Maintaining the Item Rules

You have to specify the region which contains the desired text. Text can be moved or copied to the status bar, the window title or the dialog title. It is also possible that specific regions are not shown at all.

### ▶ To display all defined Item rules

- From the **Basic** menu, choose **Item**.

The Item dialog box appears. When the Item dialog box is shown, a check mark is shown next to the **Item** command.



When items have already been defined, the defined regions are shown in this dialog box. The scope window then shows the outlines of all defined items.

The color red in the symbol to the left of each defined region indicates that this rule has been disabled. A symbol with the color green indicates that the rule is enabled.

The following command buttons are available:

<b>New</b>	Add a new Item rule (see below).
<b>Edit</b>	Modify the selected Item rule. Alternative: double-click an Item rule.
<b>Delete</b>	Delete the selected Item rule. Alternative: choose <b>Delete</b> from the <b>Edit</b> menu or press DEL to delete the selected Item rule.

▶ **To add an Item rule**

1. Make sure that the resource file that is to be used as a template is shown in the scope window.
2. Display the Item dialog box as described above.
3. Choose the **New** button.
4. In the scope window, use the mouse to select the region in which this rule is to be applied.

The Item Configuration dialog box appears.

5. Specify all required information as described below.
6. Choose the **OK** button.

## Overview of Options

### Disable this Item

When this check box is selected, this Item rule is disabled.

### Row / Column / Width

The position of the selected region. Using the spin buttons, you can manually adjust the values.

### Description

The text found in the selected region is automatically provided as a description. This helps to identify the different Item rules.

## Location

Select one of the following option buttons to specify where the text found in the defined region is to be shown:

- **Status bar**  
The text is shown at the left, center or right of the status bar, depending on the option button selected in the Alignment group box.
- **Window title**  
The text is shown in the title bar of the viewer window (instead of "Entire Screen Builder" which is shown by default).
- **Dialog title**  
The text is shown in the title bar of the viewer window. When a window title has also been defined, it is shown behind the window title.
- **Non Display**  
The text is not shown at all.

## Alignment

Only available when the option button **Status bar** is selected. Select one of the option buttons in this group box to specify whether the text is to be shown at the left, center or right of the status bar.

## Apply this item

Select one of the option buttons in this group box to define where this rule is to be applied:

- only in parent windows,
- only in child windows, or
- in both parent and child windows.

## Copy/Move

Select one of the option buttons in this group box to specify whether the text in the defined region is to be copied or moved. When the text is moved, it is no longer shown in its original position.

# Lines

Use this basic rule to convert lines in the character screens to Windows lines. For example:



This chapter covers the following topics:

- Maintaining the Lines Rules
  - Overview of Options
  - Examples
- 

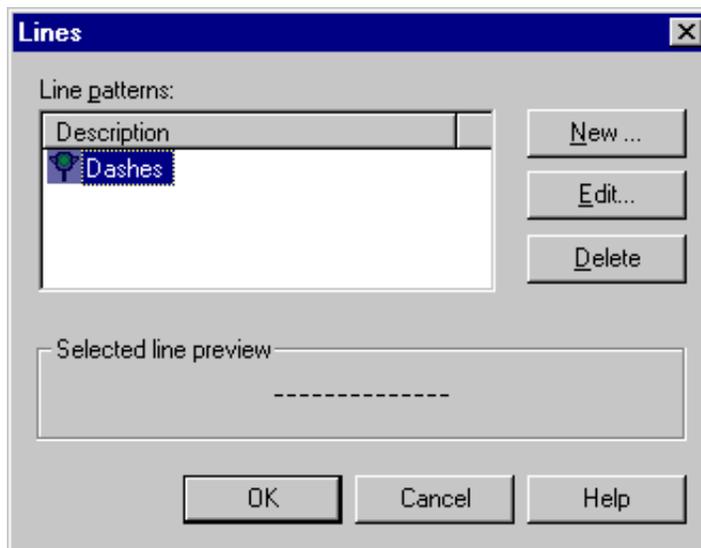
## Maintaining the Lines Rules

You have to define a pattern for each line that you want to convert to a Windows line.

### ▶ To display all defined Lines rules

- From the **Basic** menu, choose **Lines**.

The Lines dialog box appears.



When patterns have already been defined, the description of each pattern is shown in this dialog box. When you select a description, the defined pattern is shown at the bottom of the dialog box.

The color red in the symbol to the left of a description indicates that this rule has been disabled. A symbol with the color green indicates that the rule is enabled.

The following command buttons are available:

<b>New</b>	Add a new Line rule (see below).
<b>Edit</b>	Modify the selected pattern for a Line rule. Alternative: double-click a pattern for a Line rule.
<b>Delete</b>	Delete the selected Line rule.

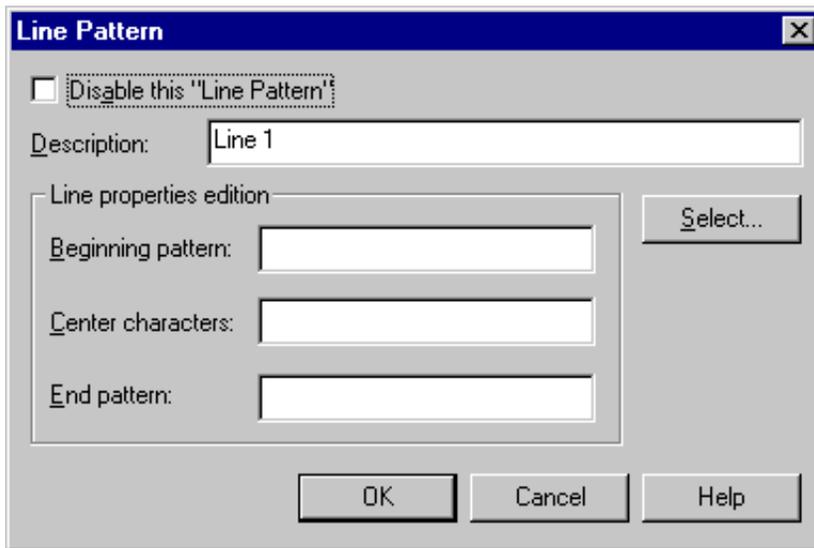
 **To add a Line rule**

1. Display the Lines dialog box as described above.
2. Choose the **New** button.

The Line Pattern dialog box appears.

3. Specify all required information as described below.
4. Choose the **OK** button.

## Overview of Options



### **Disable this Line Pattern**

When this check box is selected, this rule is disabled.

### **Description**

Enter a description for this pattern. A default description is automatically provided.

### **Beginning pattern**

Defines the character that is used in the character screen for the beginning of the line. You can enter one or more characters in this text box.

### **Center characters**

Defines the character that is used in the character screen for the center of the line. You can enter one or more characters in this text box.

### **End pattern**

Defines the character that is used in the character screen for the end of the line. You can enter one or more characters in this text box.

### **Select**

This command button invokes the Characters Selector dialog box. Instead of typing the characters in the corresponding text boxes, you can also select them from this dialog box. Before choosing a character, position the cursor in the appropriate text box.

## Examples

The following table gives examples for defining the rules for different types of line:

	<b>Example 1</b>	<b>Example 2</b>
<b>Line in character screen</b>	-----	<<=====>>>
<b>Beginning pattern</b>	-	<<<
<b>Center characters</b>	-	=
<b>End pattern</b>	-	>>>

**Tip:**

If you want to prevent dashes or hyphens to be converted to lines, enter one or more minus characters as the beginning, center or end pattern.

# Map Detection

Use this basic rule to identify a screen in which rules are to be applied in map scope. This rule is only available in application scope.

This chapter covers the following topics:

- Maintaining the Map Detection Rules
  - Overview of Options
- 

## Maintaining the Map Detection Rules

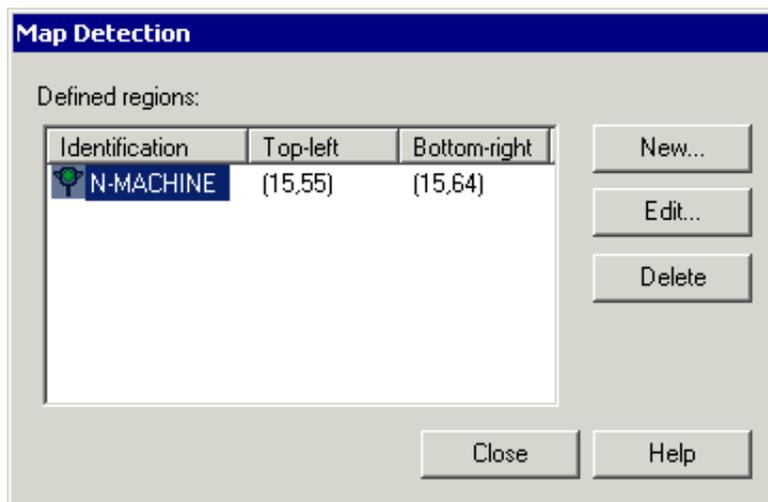
A screen is detected when a defined region is found on a host screen, or when a defined region *and* all defined subregions are found on a host screen. For each host screen, you can define more than one region.

When a screen that has been defined with the Map Detection rule is detected, the rules defined for application scope are not applied. Instead, the rules defined for map scope (i.e. for this screen) are applied.

### ▶ To display all defined Map Detection rules

- From the **Basic** menu, choose **Map Detection**.

The Map Detection dialog box appears. When this dialog box is shown, a check mark is shown next to the **Map Detection** command.



When Map Detection rules have already been defined, they are shown in this dialog box. The scope window then shows the outlines of all defined Map Detection rules.

The color red in the symbol to the left of a defined region indicates that this rule has been disabled. A symbol with the color green indicates that the rule is enabled.

The following command buttons are available:

<b>New</b>	Add a new Map Detection rule (see below).
<b>Edit</b>	Modify the selected Map Detection rule. Alternative: double-click a Map Detection rule.
<b>Delete</b>	Delete the selected Map Detection rule. Alternative: choose <b>Delete</b> from the <b>Edit</b> menu or press DEL to delete the selected Map Detection rule.

 **To add a Map Detection rule**

1. Make sure that the resource file that is to be used as a template is shown in the scope window.
2. Display the Map Detection dialog box as described above.
3. Choose the **New** button.
4. In the scope window, use the mouse to select the region which contains the string that is to be used to detect the screen.

The Map Detection Configuration dialog box appears.

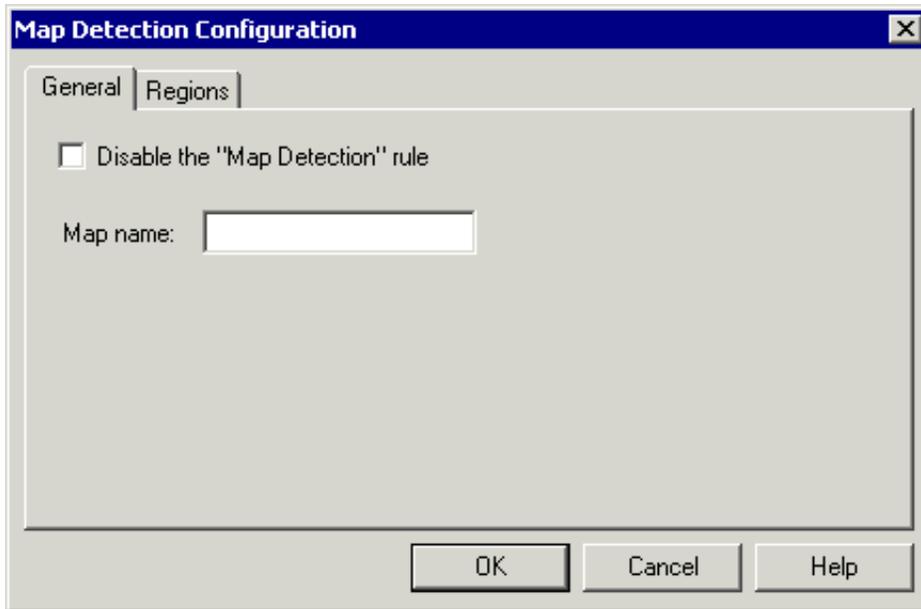
5. Specify all required information as described below.
6. Choose the **OK** button.

## Overview of Options

Different pages are available in the Map Detection Configuration dialog box:

- General
- Regions

### General



#### Disable the Map Detection rule

When this check box is selected, this Map Detection rule is disabled.

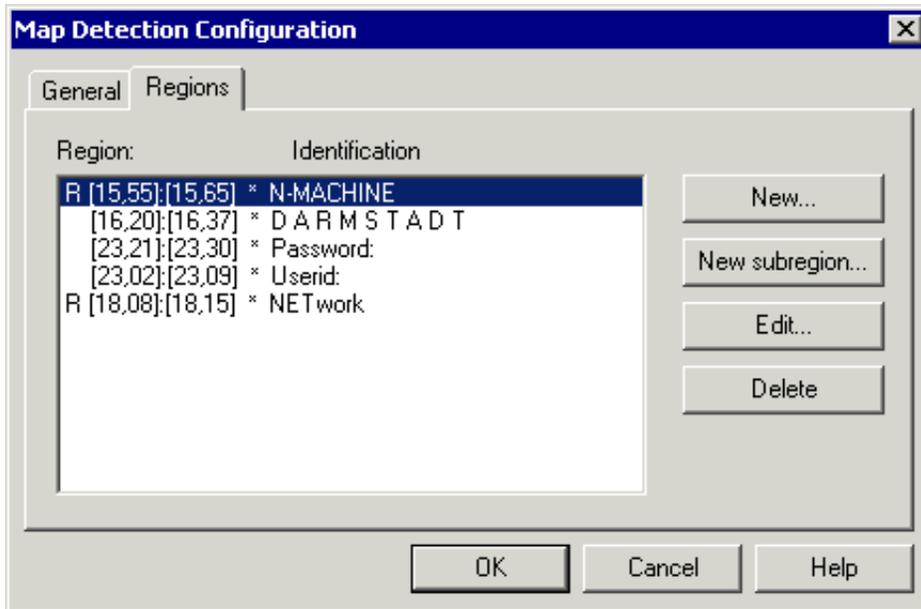
#### Map name

Specify the name that is to be used for map scope. The name can be up to 8 characters long. An *ini* file with this name is created in the folder that has been created for application scope. This map name is then available in the Open Scope dialog box. See *Using Different Scopes* in the documentation *Defining the Rules Using the SDK* for further information.

You can close the current application scope and open map scope. All rules that you define in map scope are written to the *ini* file that has been created for this map.

## Regions

The Regions page is used to define regions and subregions, i.e. unique strings which appear on the host screen that is to be detected.



On this page, a defined region is indicated by the letter "R" at the beginning of a line. Lines which do not begin with the letter "R" apply to subregions.

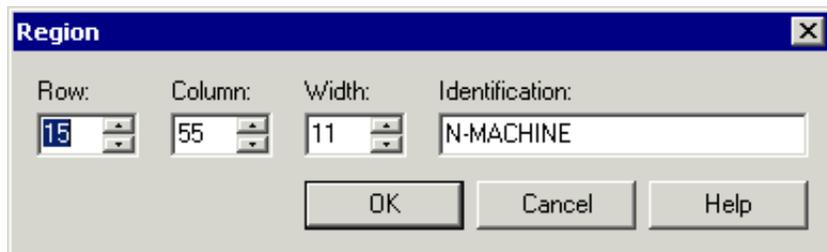
Example: with the above definitions, the screen is detected when region 1 *and* all defined subregions are found, *or* when region 2 is found.

The following command buttons are provided:

<b>New</b>	Define a new region for the current map.
<b>New subregion</b>	Define a new subregion for the selected region. For one region several subregions can be defined.
<b>Edit</b>	Modify the selected region or subregion. Alternative: double-click a region or subregion to modify it. See <i>Modifying a Region or Subregion</i> below.
<b>Delete</b>	Delete the selected region or subregion. When you delete a region, all of its subregions are automatically deleted.

## Modifying a Region or Subregion

The Region dialog box appears when you select a region or subregion on the Regions page and choose the **Edit** button (or when you double-click it).



### Row / Column / Width

The position of the string that is to be used to detect the screen. This is the region that has been selected using the mouse. Using the spin buttons, you can manually adjust the values.

### Identification

This is the string that has been selected using the mouse. This string is used to detect the screen on which the rules for map scope are to be applied.

# Message Line

Use this basic rule to display the message line in the status bar or in a specific line of the dialog.

**Important:**

If the message line contains an input field, it is not moved to another position.

On BS2000 hosts there may be an input field at the end of the message line. To remove this input field, use the Natural commands %t=9756 and %KN. To change this back, use %t=9750 and %KS.

This chapter covers the following topics:

- Maintaining the Message Line Rule
- Overview of Options

## Maintaining the Message Line Rule

The text from the message line can be moved to the status bar or to a specific line of the dialog.

The viewer needs to know where the message line is shown in the character screen so that it can be moved to the status bar or a specific line in the dialog.

**Example 1 - The message line appears in the bottom of the main screen:**

```

SOFTWARE AG      *** EMPLOYEE MANAGEMENT ***      21/04/99
NSWEMPAL        ADD EMPLOYEES                      17:45:09
-----
PERSONAL DATA

CODE.....: _____
NAME.....: _____ SURNAME.....: _____
MARITAL ST:___ DATE OF BIRTH..:___ __ __

ADDRESS

ADDRESS...: _____ CITY.....: _____
COUNTRY...:___ POST CODE.: _____

COMPANY DATA

DEPARTMENT.....: _____
CATEGORY.....: _____

CODE field is mandatory.
    
```

**Example 2 - The message line appears in the bottom of a child window (help routine):**

```

SOFTWARE AG      *** EMPLOYEE MANAGEMENT ***      21/04/99
NSWEMPA1        ADD EMPLOYEES                      17:45:09
-----

PERSONAL DATA                                     +-----+
CODE.....:_____                               | 1.- GER |
NAME.....:_____ SURNAME.....                 | 2.- SPA |
MARITAL ST:___ DATE OF BIRTH                   | 3.- UK  |
                                                | 4.- USA |
ADDRESS                                           | Option: ___ |
ADDRESS...:_____ CITY.....:_____ | Select country |
COUNTRY...:?___ POST CODE.:_____             +-----+

COMPANY DATA

DEPARTMENT.....:_____
CATEGORY.....:_____
    
```

**Example 3 - The message line appears in the bottom of the main screen (help routine):**

```

SOFTWARE AG      *** EMPLOYEE MANAGEMENT ***      21/04/99
NSWEMPA1        ADD EMPLOYEES                      17:45:09
-----

PERSONAL DATA                                     +-----+
CODE.....:_____                               | 1.- GER |
NAME.....:_____ SURNAME.....                 | 2.- SPA |
MARITAL ST:___ DATE OF BIRTH                   | 3.- UK  |
                                                | 4.- USA |
ADDRESS                                           | Option: ___ |
ADDRESS...:_____ CITY.....:_____ |
COUNTRY...:?___ POST CODE.:_____             +-----+

COMPANY DATA

DEPARTMENT.....:_____
CATEGORY.....:_____

Select country.
    
```

### ▶ To define the Message Line rule

1. From the **Basic** menu, choose **Message Line**.

The Message Line dialog box appears.

2. Specify all required information as described below.
3. Choose the **OK** button.

## Overview of Options



### Disable the Message Line rule

When this check box is selected, the Message Line rule is disabled.

### The message line always appears in the main screen

When this check box is selected, the viewer assumes that the message line is always shown in the main screen. When this check box is not selected, the viewer gets the message line from the last detected main screen or child window.

### Get line from character screen

So that the message line can be detected, you have to specify its location in the character screen. Select one of the following option buttons:

- **Top**  
The message line is located in the first line of the screen.
- **Bottom**  
The message line is located in the 24th (i.e. the last) line of the screen.
- **Other**  
Use the spin box to specify the number of any other line (1 to 24).

Under Natural UNIX and Natural OpenVMS, the message line is detected automatically if the environment variable `NSW_PF_MSG_LINES_NATIVE_FORMAT` has not been set to "YES". See the sections *Installing Entire Screen Builder on Natural UNIX Hosts* and *Installing Entire Screen Builder on Natural OpenVMS Hosts* in the *Installation and Configuration* documentation for further information on this environment variable.

### Display line in output window

Select one of the following option buttons to specify where the message line is to appear:

- **Status bar**  
The information from the message line is shown in the status bar of the viewer. Use the drop-down list box to specify whether the message line is to appear at the left, right or center of the status bar.
- **In dialog line**  
The information from the message line is shown in a specific line of the dialog. Use the spin box to specify the number of the line (1 to 24). If you want to define another font and/or color for the message line, choose the **Font** button and specify all required options in the resulting Font dialog box.

# User Exit

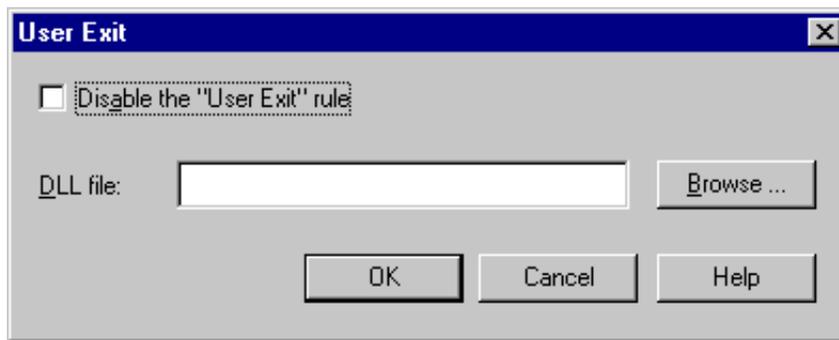
Use this basic rule to define a DLL containing user exits.

When a user exit has been defined, the viewer will then call the functions `NSWBefore` and `NSWAfter` for each received and sent screen. See the *User Exits* documentation for further information.

## ▶ To define the DLL containing the user exits

1. From the **Basic** menu, choose **User Exit**.

The User Exit dialog box appears. When a DLL has already been defined, the corresponding path is shown in this dialog box.



2. To define a DLL, specify the path to the DLL in the text box, or choose the **Browse** button to select the DLL from the Open dialog box..

Or:

If the defined DLL is no longer to be used, select the **Disable the User Exit rule** check box.

3. Choose the **OK** button.