Entire Screen Builder

	Version 5.2.1	Defining the Rules Using the SDK

software AG

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

Defining the Rules Using the SDK . . . 1
Defining the Rules Using the SDK . . . 1
Getting Started . . . 2
Getting Started . . . 2
Starting the SDK . . . 2
Elements of the Application Window . . . 3
Menu Bar . . . 4
Toolbar . . . 5
Font Toolbar . . . 6
Scope Window . . . 7
Status Bar . . . 8
Output Window . . . 8
Using Help . . . 9
Quitting the SDK . . . 9
Working with the SDK . . . 10
Working with the SDK . . . 10
Configuring Entire Screen Builder . . . 11
General Information on Image Files . . . 12
Using Different Scopes . . . 13
Opening a Scope . . . 15
Saving the Current Scope . . . 16
Closing the Current Scope . . . 16
Working with Host Sessions . . . 17
Defining the Client Control Properties . . . 17
Starting a Host Session . . . 21
Terminating a Host Session . . . 21
Using Screen Files . . . 22
Capturing a Screen File . . . 22
Saving a Host Screen as a Resource File . . . 23
Capturing and Saving a Screen File Automatically . . . 24
Opening a Screen File . . . 25
Defining an Application . . . 27
Modifying a Region or Subregion . . . 30
Example for Regions and Subregions . . . 31
Defining a Help File for an Application . . . 31
Defining the Last Rules Used . . . 32
Zooming the Contents of the Scope Window . . . 33
Rules Analysis . . . 33
Making the Rules Available for All Users . . . 35
Defining Basic Rules . . . 36
Defining Basic Rules . . . 36
Overview of Tasks and Corresponding Basic Rules . . . 36
General Information on the Rules . . . 38
Using Basic Rules Together with Extended Rules . . . 41
Showing Images and Fields . . . 43
Defining Regions for Specific Rules . . . 44
Displaying Help Information when Defining Regions . . . 45

Defining Extended Rules 46
Defining Extended Rules 46
Using a Resource Editor 46
Creating a DLL 47
Reserved Control IDs 52
Reserved Dialog Names 52
Opening a DLL 53
Opening a Dialog 53
Closing a Dialog 54
Defining the Dialog Properties 55
Defining the Control Properties 57
Field Identification 58
Display Styles 60
Action 61
Mouse 64
Screen Number 65
Defining the Font 66
Using Styles 67
Defining a Control Style 67
Defining a Dialog Style 72
Applying a Style to a Control or Dialog 75
Saving the DDT File 75
Building the BDD File 76
Detecting the Screen on which the Extended Rules are to be Applied (Single Screen) . . . 78
Overview of Extended Detection Configuration Options 79
Using the Multi Screen Feature 83
Using the Multi Screen Feature 83
Multi Screen Scenarios 83
Navigation 84
Output Collection from Several Screens (Many to One) 85
Input to Many Screens (One to Many) 86
Required Steps 87
Displaying All Defined Multi Screen Rules 88
Detecting the First Character Screen 89
Defining the BDD File and the Dialog 90
Defining the Green Screens 91
Defining the Regions and Subregions for Each Screen 93
Defining the Options for Each Screen 95
Sending Data to an Input Field on the Host 96
Error Handling 97

Defining the Rules Using the SDK

The Entire Screen Builder SDK (Software Development Kit) allows you to define basic and extended transformation rules. These rules will then be applied when you view the application using one of the Entire Screen Builder GUI viewers.

This documentation is organized under the following headings:

Getting Started	How to start and quit the SDK, information on the elements of the application window, and how to access help for the Entire Screen Builder.
Working with the SDK	How to define the folder containing the transformation rules, store images, use different scopes, work with host sessions, capture a screen file to be used as a template for defining rules, add different applications for defining different rules, define a help file for an application, define the type of rules (global or application) to be applied to a character screen for which an application has not been defined, zoom the contents of the scope window, display the current rules trace file for rules analysis, and copy the rules to the production environment.
Defining Basic Rules	An overview of tasks (for example, move text from the character screen to the status bar) and corresponding basic rules. Information on the default behavior of each rule, on the scope and type of window in which a specific rule can be defined, and how basic rules interact with extended rules. How to show images in the scope window and the fields that have been defined on the character screen, and how to define regions for specific rules.
Defining Extended Rules	How to create a DLL using a resource editor. Reserved control IDs and dialog names. How to open a DLL and its dialogs in the SDK, apply extended rules to the controls in a dialog, define a background color, style or image for a dialog, apply styles to controls and dialogs, save the definitions for the dialog controls, and associate a dialog with the character screen for which it is to be displayed.
Using the Multi Screen Feature	How to define a dialog containing extended rules for more than one character screen (for example, how to collect data from several character screens and show this data in a single dialog).

Getting Started

This chapter tells you how to get started with the Entire Screen Builder SDK. The following topics are provided:

- Starting the SDK
- Elements of the Application Window
- Using Help
- Quitting the SDK

Starting the SDK

If you do not specify otherwise during installation, an Entire Screen Builder folder automatically appears in the Programs folder of the Start menu after Entire Screen Builder has been installed. It contains the shortcuts for the Entire Screen Builder components, including the SDK.

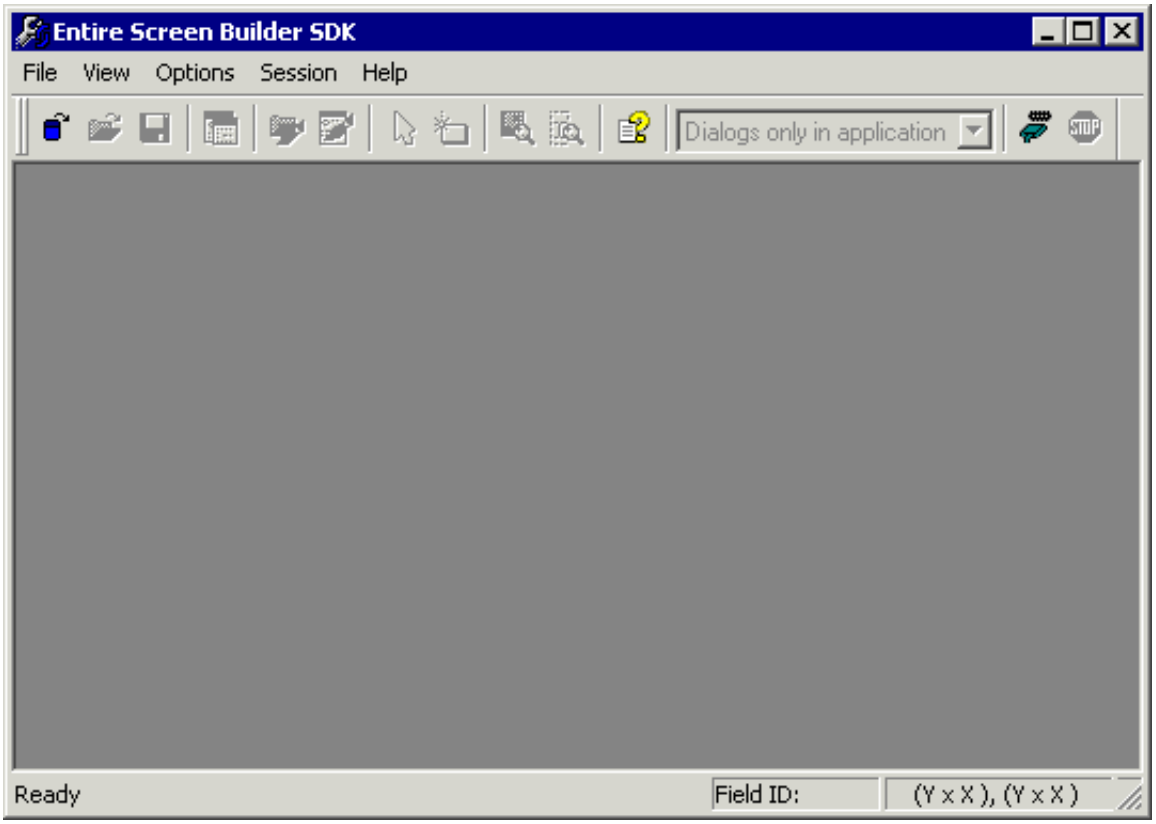
To start the SDK

- From the Start menu, choose **Programs** > **Software AG Entire Screen Builder *n.n.n*** > **SDK**.

  The SDK is now started.

Elements of the Application Window

Directly after starting the SDK, only the menus **File**, **View**, **Options**, **Session** and **Help** are available. After opening a scope, additional menus are available.



The following topics are covered below:

- Menu Bar
- Toolbar
- Font Toolbar
- Scope Window
- Status Bar
- Output Window

Menu Bar

When a scope has been opened, the following menus are available:

Menu	Using the commands in this menu, you can ...
File	Manage scopes and screen files, open a DLL, save a DDT file, capture screens or define a new application.
Edit	Manage the regions which have been defined for specific basic rules.
View	Show or hide the various elements of the application window (such as the toolbar, status bar, defined images and fields, or a dialog box with help information on how to create a region) and zoom the contents of the scope window.
Basic	Define the different basic rules.
Extended	Define the screens on which extended rules are to be applied, or open a dialog contained in a DLL, define the control properties and dialog properties, and build the BDD file.
Options	Define the rules folder and specify which rules are to be used when no application is detected.
Session	Start and terminate a host session, or define client control properties such as the server address.
Help	Invoke the online documentation or access Software AG web sites.

Toolbar

You can execute the most important SDK functions using the toolbar.

Using the mouse, you can drag the toolbar to another position:

- so that it is shown to the left, right or bottom of the application window, or
- so that it is shown in a window of its own:

You can move the window freely on your screen. You can move it back to the application window (e.g. back to its original position below the menu bar) so that it is no longer shown in a window. This process is called "docking". To prevent docking, press CTRL while moving the window.

The toolbar buttons represent the following menu commands:

	Open Scope (**File** menu)
	Open Screen File (**File** menu)
	Save Scope (**File** menu)
	Applications Detection (**File** menu)
	Open DLL (**File** menu)
	Open Dialog (**Extended** menu)
	Select Region (**Edit** menu)
	New Region (**Edit** menu)
	Show Images (**View** menu)
	Show Fields (**View** menu)
	Online Documentation (**Help** menu)
	Connect (**Session** menu)
	Disconnect (**Session** menu)

When a DLL file has been opened, the drop-down list box in the toolbar provides for selection all dialogs defined in this DLL.

To switch the toolbar display on and off

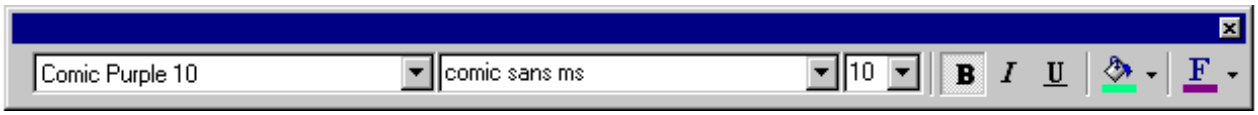
- From the **View** menu, choose **Toolbar**.

  When the toolbar is displayed in the application window, a check mark is shown next to this menu command.

Font Toolbar

The font toolbar is only available when a dialog with extended rules is shown. It is used to specify the font and color for the controls in the dialog.

Like the regular toolbar (see the above description) it can be dragged to another position or shown in a window of its own.



Use this toolbar to define font and/or color for the selected control as with any other Windows application. The first drop-down list box provides all styles that you have defined (see *Using Styles*). The last two buttons are used to define the background color and text color of the selected control.

See also: *Defining the Font*.

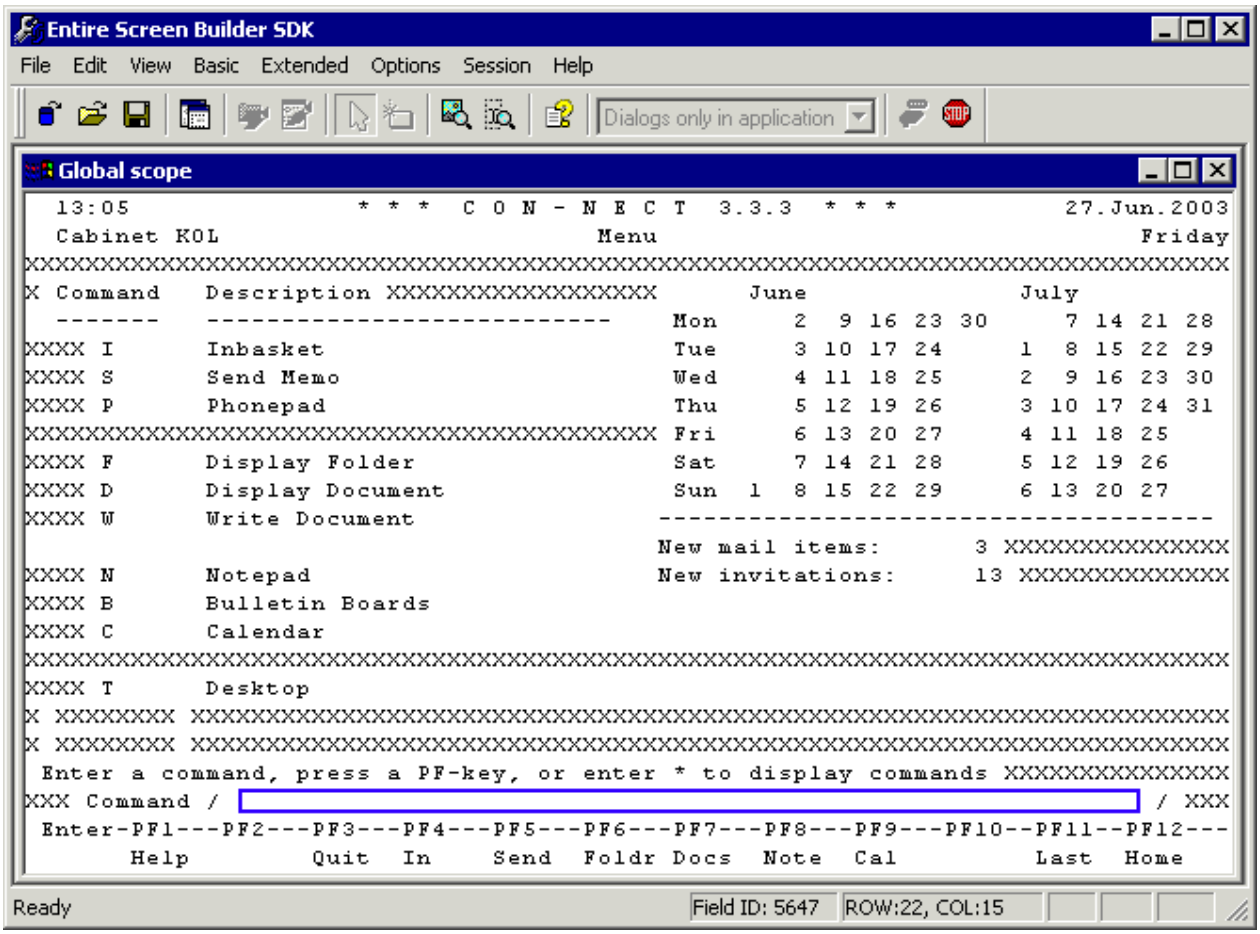
To switch the font toolbar display on and off

- From the **View** menu, choose **Font Toolbar**.

  When the font toolbar is displayed in the application window, a check mark is shown next to this menu command.

Scope Window

When you open a scope, the screen file that was last open in this scope is automatically shown in the scope window. When a screen file is shown, you can apply rules to it.



It is also possible to define rules when no screen file is shown in the scope window. In this case, you use the numbers that are shown for the rows and columns for your orientation. However, it is much harder to define the rules this way. Therefore, it is recommended that you always use a screen file as a template.

The title bar of the scope window indicates which scope is currently active:

- When global scope is active, "Global scope" is shown.
- When application scope is active, "Application:" and the name of the current application are shown.
- When map scope is active, "Application:", the name of the current application, and the name of the current map in square brackets are shown.

See *Using Different Scopes* for further information.

Status Bar

The status bar is used to display system messages and help texts for the currently selected menu command or toolbar button.

When you move the mouse over the scope window, the status bar also shows the current mouse position (row and column) and the current field ID (to the left of the current mouse position).

When you move the mouse over a dialog with extended rules, the status bar also shows the ID of the current control and type of control. This information is shown at the very left of the status bar.

To switch the status bar display on and off

- From the **View** menu, choose **Status Bar**.

  When the status bar is displayed in the application window, a check mark is shown next to this menu command.

Output Window

An output window appears when you build the BDD file for a dialog with extended rules. It informs you whether the compilation of the BDD was successful. When compilation was not successful, all detected errors are shown in the output window.

For further information, see *Building the BDD File*.

To switch the output window display on and off

- From the **View** menu, choose **Output Window**.

  When the output window is displayed in the application window, a check mark is shown next to this menu command.

Using Help

The complete Entire Screen Builder documentation is available as an HTML help file.

To access the documentation

- From the **Help** menu, choose **Online Documentation**.

  Or:
  Choose the following toolbar button:

To invoke context-sensitive help

- In a dialog box, choose the **Help** button.

Quitting the SDK

When you quit the SDK and your changes have not yet been saved, you are asked whether you want to save them.

To quit the SDK

- From the **File** menu, choose **Exit**.

  Or:
  Choose **Close** from the Control menu.

  Or:
  Choose the corresponding standard button in the title bar.

Working with the SDK

This chapter covers the following topics:

- Configuring Entire Screen Builder
- General Information on Image Files
- Using Different Scopes
- Working with Host Sessions
- Using Screen Files
- Defining an Application
- Defining a Help File for an Application
- Defining the Last Rules Used
- Zooming the Contents of the Scope Window
- Rules Analysis
- Making the Rules Available for All Users

For further information, see *Defining Basic Rules* and *Defining Extended Rules*.

Configuring Entire Screen Builder

When you configure Entire Screen Builder, you define the folder in your local development repository which contains the transformation rules that are to be applied.

It is only possible to configure Entire Screen Builder when no scope has been opened.

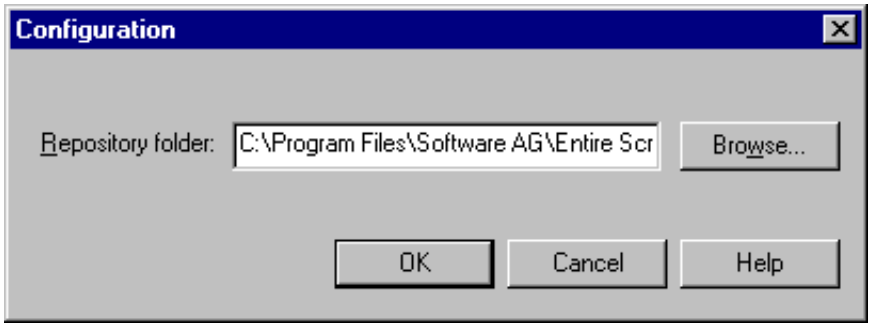
You can also define the location of the rules repository with the System Management Hub. See *GUI Version* in Entire Screen Builder's *System Management Hub* documentation.

Important:
When another folder is defined with the System Management Hub and the SDK is still active, this change is only applied to the SDK after the SDK has been started once more.

To define the folder containing the transformation rules in the SDK

1. From the **Options** menu, choose **Configuration**.

   The Configuration dialog box appears.



2. Specify the path to the folder containing all defined transformation rules.

   You can also choose the **Browse** button to select another folder from the Browse for Folder dialog box.

3. Choose the **OK** button to save your specifications.

   A dialog appears asking whether you want to restart the Entire Screen Builder Server. A restart is required if you want to use the new repository immediately.

4. If you want to use the new repository, choose the **Yes** button.

General Information on Image Files

Make sure that all image and cursor files that you define with the SDK are stored in the rules repository. The SDK stores the files with relative path names - relative to the current repository setting.

Caution:
Do not use absolute path names for your image files and cursors. When using absolute path names, your files will not be found.

In a development environment, an error message appears when an image cannot be found. It is then up to the developer to take further action. In a production environment, such an error message is not shown.

Note on image formats:

- The GUI viewers support the following formats: GIF, JPG and BMP.
- Animated GIF files are not supported.

Using Different Scopes

How your transformation rules are applied, depends on the scope for which they are defined.

The transformation rules are structured in scopes, where each scope can have its own set of transformation rules. A scope is entered when the screen detection defined for this scope is successful.

Scope detection is performed each time a new screen is received from the host. It works in this way:

- Application detection is performed starting with the first application detection rule defined in the repository.
- If an application detection is successful for the screen, the map detections in this application scope are performed.
    - If a map detection is successful, the rules from the corresponding map scope are used to display the screen.
    - If a map detection is not successful, the rules from the application scope are used.
- If none of the application detections is successful, then, depending on the setting of the **Last Rules Used** command (see *Defining the Last Rules Used*), the rules from the application scope detected for the previous screen or the rules from the global scope are used.

Scope	Description
Global	The rules are applied to a screen each time no other scope is detected.
Application	The rules are applied to all screens of a specific application. They overwrite the rules defined for global scope. You have to define the application in global scope. See *Defining an Application* for further information.
Map	The rules are applied to a specific screen. They overwrite the rules defined for application scope. Map scope enables you to define transformation rules for screens that do not follow the general application standards. In this case, you have to define the Map Detection rule in application scope. This rule defines how to identify a particular screen.

This hierarchical structure provides a very flexible method of defining rules because different transformation rules may be applied to different applications that are not written with the same screen definitions (for example, when the applications have different function key representations).

Important:
Rules are not inherited between the different scopes.

Basic rules can be applied in global, application and map scope. A basic rule defines, for example, that the name of the application is to appear in the title bar of the application window, or that the content of an output field is to be shown in red color.

Extended rules can only be applied in application scope and map scope. They refer to dialogs that have been created with a resource editor. An extended rule can be used, for example, to substitute input fields with radio buttons.

For each scope you define, a configuration file containing the corresponding rules is available in your local repository. By default, the local repository is stored in the Entire Screen Builder folder *Repository*.

- In global scope, all rules are written to the *Defaults.ini* file in the folder *Repository*.
- In application scope, all rules are written to the *Defaults.ini* file of the folder that has been created for this application.
- In map scope, all rules are written to the *mapname.ini* file of the folder that has been created for the application in which the Map Detection rule has been defined.

The following information is provided below:

- Opening a Scope
- Saving the Current Scope
- Closing the Current Scope

Opening a Scope

Only one scope can be open at one time. Opening a scope automatically closes the current scope; if you have not yet saved your modifications, you are prompted to save them.

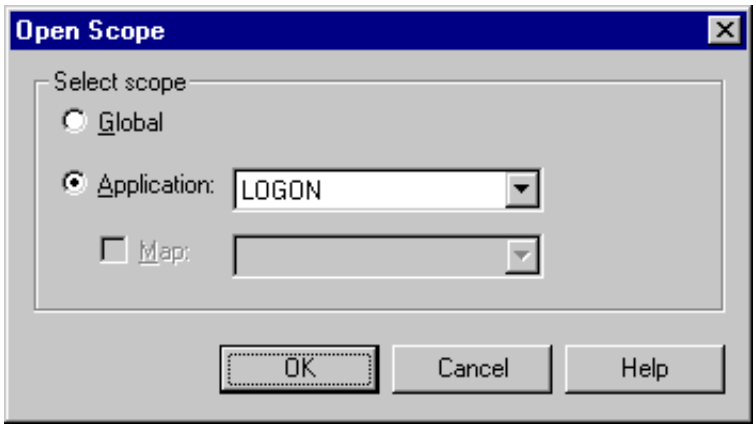
Important:
Before you start defining rules, make sure that the required scope has been opened.

To open a scope

1. From the **File** menu, choose **Open Scope**.

   Or:
   Choose the following toolbar button:

   The Open Scope dialog box appears.

2. Select the desired scope: global or application.

   As long as an application has not been defined, only global scope is available (see *Defining an Application* later in this section).

3. If you have selected application scope, select the desired application from the **Application** drop-down list box.

   When the Map Detection rule has been defined for the selected application, all defined maps are provided in the **Map** drop-down list box. In this case, it is possible to work in map scope.

4. Optionally - if you want to work in map scope, select the **Map** check box and select the desired map from the corresponding drop-down list box.

5. Choose the **OK** button.

   A window for the selected scope is shown. If a screen file has previously been opened in this scope, it is automatically shown.

Saving the Current Scope

When you save the rules for the current scope, the *Defaults.ini* file in the corresponding folder is updated.

To save the scope

- From the **File** menu, choose **Save Scope**.

  Or:
  Choose the following toolbar button:

Closing the Current Scope

When you close the current scope and have not yet saved your modifications, you are prompted to save them.

To close the scope

- From the **File** menu, choose **Close Scope**.

  Or:
  Close the scope window by choosing the corresponding standard button in its title bar.

Working with Host Sessions

You can display the host screens using the transformation rules that you have defined. This is helpful during the development cycle in order to save host screens as screen files and to check how the defined rules are applied.

The following information is provided below:

- Defining the Client Control Properties
- Starting a Host Session
- Terminating a Host Session

Defining the Client Control Properties

In order to open a host session in the Windows Viewer, you have to define the client control properties, i.e. specify all necessary parameters. This information is written to the Windows registry.

For the Web Viewer, these properties are set in the HTML page.

To define the client control properties

1. From the **Session** menu, choose **Properties**.

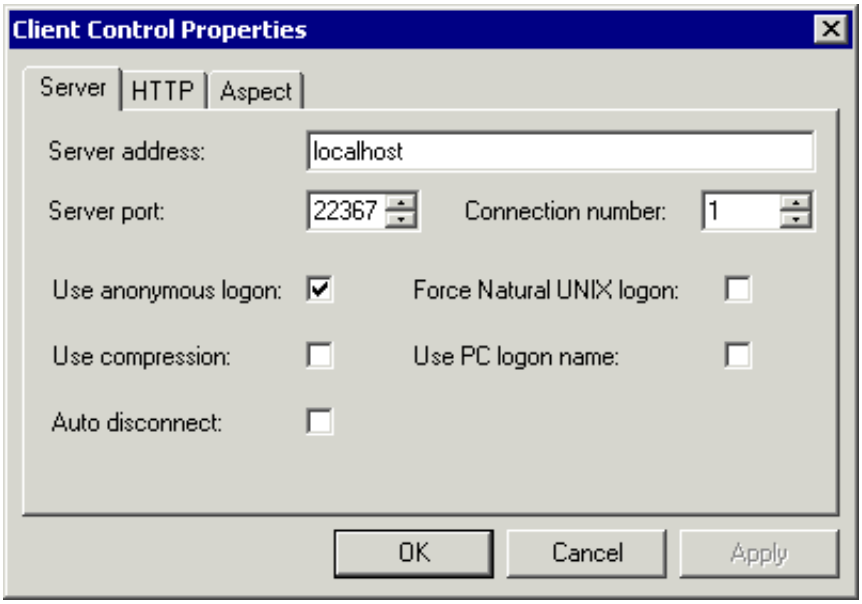
   The Client Control Properties dialog box appears. You can define properties on the following pages: Server, HTTP and Aspect. Information on the properties is provided below.

2. Specify at least the following properties: server address, server port and connection number.

3. Choose the **OK** button to save your specifications.

Server

The settings on the Server page are used by the viewer to establish the connection with the Entire Screen Builder Server.



A detailed description for these properties can be found in the *Overview of Client Control Properties* which is part of the *User Exits* documentation.

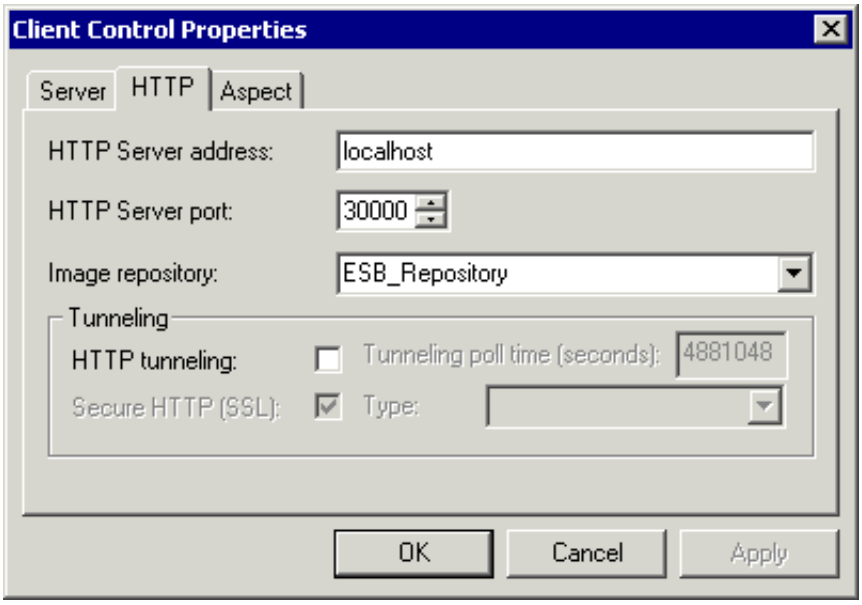
Property names in above dialog box	**Property names for user exits**
Server address	`Server`
Server port	`Port`
Connection number	`Connection`
Use anonymous logon	`AnonymousLogon`
Force Natural UNIX logon	`UnixLogon`
Use compression	`Compressed`
Use PC logon name	`UsePCLogonName`
Auto disconnect	`AutoDisconnect`

HTTP

The settings on the HTTP page are used by the viewer to get the images from the repository.

The Entire Screen Builder Server has an internal image server which can be used to get the images. Other Web servers (HTTP servers) can also be used for this purpose. In this case, the properties **HTTP server address** and **HTTP server port** must contain the correct values for the Web server.

Important:
The internal image server is not a standard Web server. It only supports the request GET (not all functionality). Do not use this server as your Web server because this will not work. We recommend to always enter the server name.



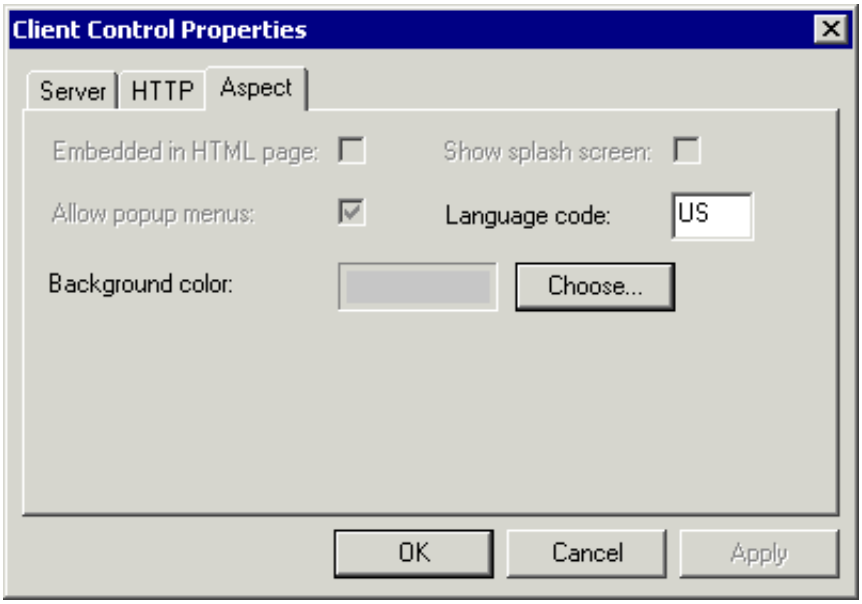
Note:
The drop-down list box **Image repository** contains all aliases defined in the System Management Hub. See *Alias List* in Entire Screen Builder's *System Management Hub* documentation.

A detailed description for these properties can be found in the *Overview of Client Control Properties* which is part of the *User Exits* documentation.

Property names in above dialog box	Property names for user exits
HTTP server address	`HttpServer`
HTTP server port	`HttpPort`
Image repository	`Repository`
HTTP tunneling	`Tunneling`
Tunneling poll time	`TunnelingPollTime`
Secure HTTP (SSL)	`SecureHttp`
Type	`TunnelingType`

Aspect

The settings on the Aspect page are used by the viewer to show the dialogs or screens received from the Entire Screen Builder Server with a particular aspect.



A detailed description for these properties can be found in the *Overview of Client Control Properties* which is part of the *User Exits* documentation.

Property names in above dialog box	Property names for user exits
Embedded in HTML page *	`Embedded`
Show splash screen *	`ShowSplash`
Allow popup menu *	`AllowPopupMenu`
Language code	This property is used to support different resource language libraries. As long as you are using the English resource libraries, as delivered with Entire Screen Builder, it is important that you do not delete the entry "US" in this text box. If you require your own resource language libraries, contact your Software AG representative.
Background color	`Background`

* Always disabled for the Windows Viewer.

Starting a Host Session

You can connect to the session (connection number) that has been defined in the client control properties.

Important:
A host session can only be started when the Entire Screen Builder Server is running.

In a development environment (which includes the SDK), you can only connect one GUI viewer to the Entire Screen Builder Server. This is to enable reloading and testing of the rules during development. This restriction does not apply to the Terminal Viewer.

For information on how to start the Entire Screen Builder Server and how to define host sessions, see Entire Screen Builder's *System Management Hub* documentation.

To start a host session

- From the **Session** menu, choose **Connect**.

  Or:
  Choose the following toolbar button:

  The session is now started in the Windows Viewer. The transformation rules that have been defined are automatically applied. See the *GUI Viewers* documentation for further information.

Terminating a Host Session

When you terminate a host session, the Windows Viewer is closed.

To terminate a host session

- From the **Session** menu of the SDK, choose **Disconnect**.

  Or:
  Choose the following toolbar button:

  Or:
  From the **File** menu of the Windows Viewer, choose **Disconnect**.

Using Screen Files

You can use a screen file as a template for defining the required rules.

The following information is provided below:

- Capturing a Screen File
- Saving a Host Screen as a Resource File
- Capturing and Saving a Screen File Automatically
- Opening a Screen File

Capturing a Screen File

You can capture the screen which is currently shown for the host session, i.e. the screen that is shown in the Windows Viewer which you have invoked using the **Connect** command (see *Starting a Host Session*). The screen is then shown in the scope window for the current scope.

You can only capture the original character screen on which the transformation rules are not applied.

To capture a character screen

1. Use the Windows Viewer to access the character screen you want to save.
2. From the **File** menu of the SDK, choose **Capture Screen**.

   The character screen is now shown in the scope window and you can save it as described below.

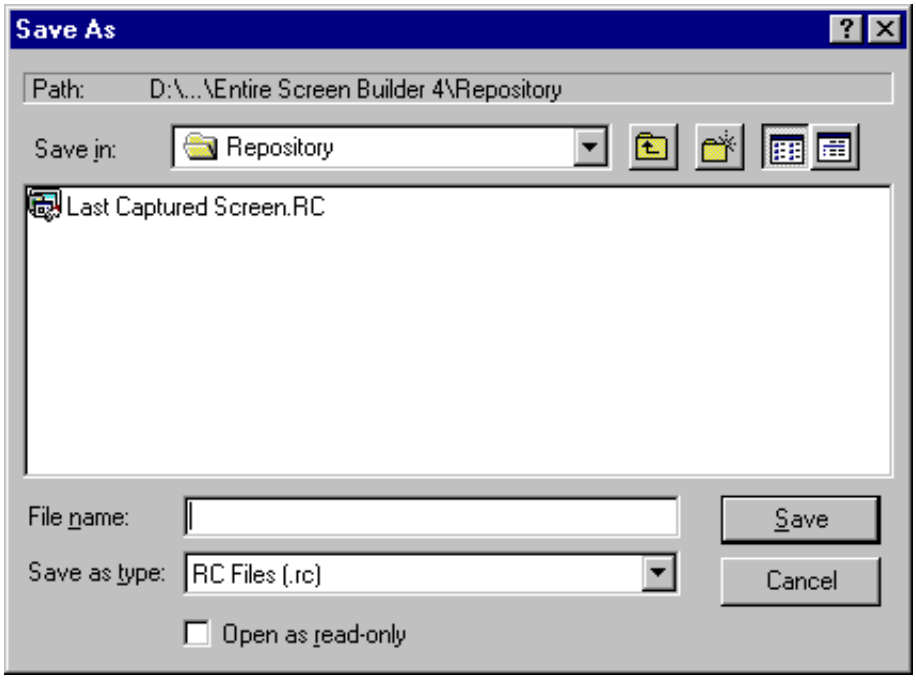
Saving a Host Screen as a Resource File

You can save the character screen that is shown in the scope window as a resource file (also called screen file).

To save a character screen as a screen file

1. From the **File** menu, choose **Save Screen File**.

   The Save As dialog box appears.



2. Specify a name for your screen file.

   You need not specify the extension *rc* since it is automatically provided.

3. Choose the **Save** button.

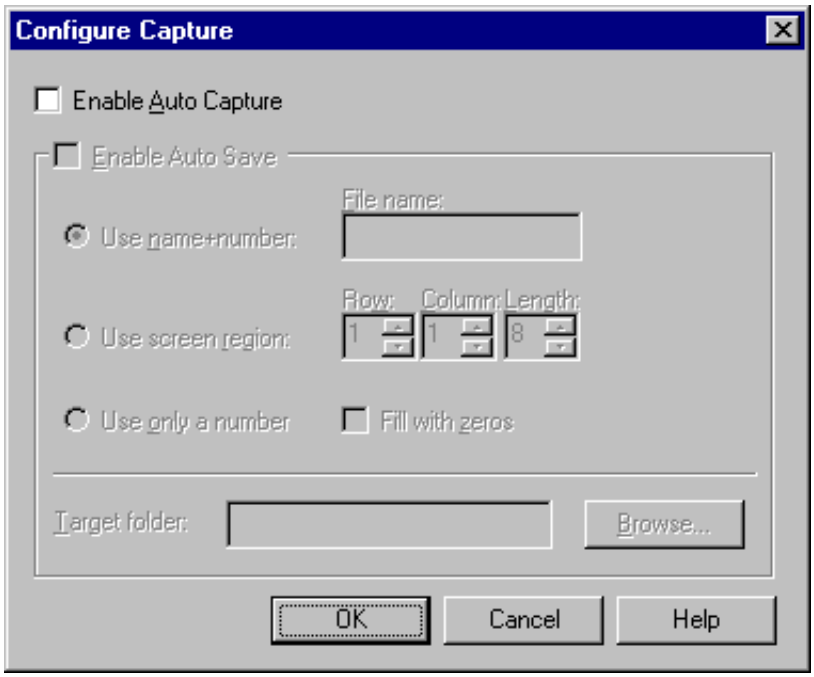
Capturing and Saving a Screen File Automatically

It is possible to capture and save each screen that you access automatically.

To capture and save automatically

1. From the **File** menu, choose **Configure Capture**.

   The Configure Capture dialog box appears.



2. Enable the **Enable Auto Capture** check box.

   As long as the **Enable Auto Save** check box is not selected, host screens cannot be saved.

   When a scope has been opened and the **Enable Auto Capture** check box has been enabled, all screens received from the host are automatically shown in the scope window, but are not saved as screen files. If you want to save these screens, enable the **Enable Auto Save** check box (see the description below). The auto-capture function can be used, for example, to define all basic rules without saving any screens.

3. When you enable the **Enable Auto Save** check box, all host screens are saved automatically. You can then select one of the following option buttons:

   Use name + number

   When you select this option, you have to specify a file name. A number that is incremented by 1 for each new file is automatically added to the file name.

Use screen region

When you select this option, you have to specify the position (row, column and length) in the character screen which contains the string that is to be used as the file name. This can be, for example, a unique map number shown in each screen.

Use only a number

When you select this option, a number that is incremented by 1 is used for each new file. You can optionally specify that the positions before the number are filled with zeros.

The file name can be up to 8 characters long.

4. In the **Target folder** text box, specify the name of the folder into which the screen files are to be saved. When you only specify a dot (.) in this text box, all screen files are saved one level above the current folder.

   You can also choose the **Browse** button to select a folder from a dialog box.

5. Choose the **OK** button.

Opening a Screen File

A screen file is a resource file that has been saved as described above.

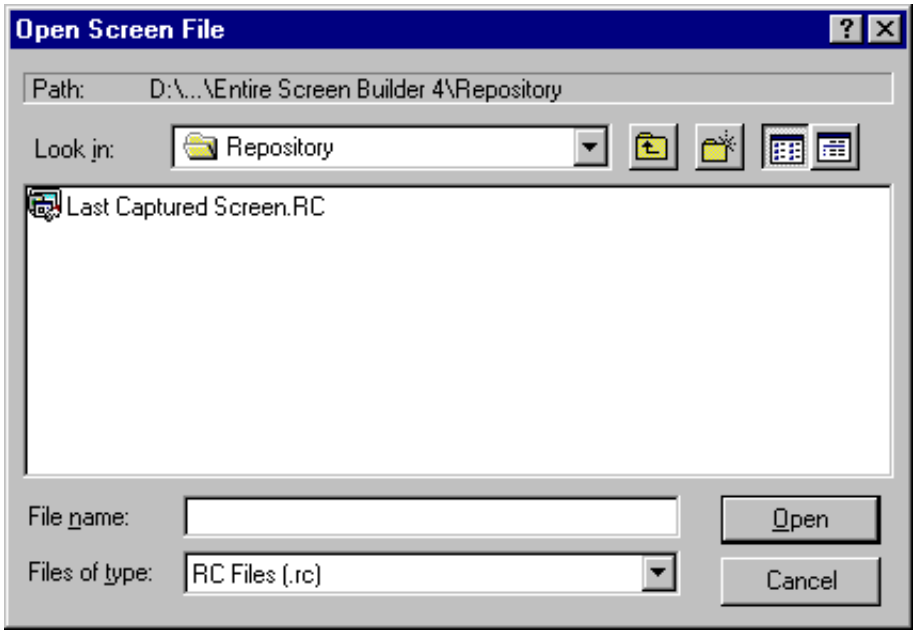
Important:
You can only open a screen file when a scope has been opened.

To open a screen file

1. From the **File** menu, choose **Open Screen File**.

   Or:
   Choose the following toolbar button:

   The Open Screen File dialog box appears. A screen file has the extension *rc*.



2. Select the desired screen file.
3. Choose the **Open** button.

   The content of the screen file is now shown in the scope window.

Defining an Application

You can define different applications. For each of these applications, you can define different rules.

The term "application" does in this case not refer to a program, but to a set of rules that is to be applied to certain screens of a program. For one program, several applications can thus be defined.

Important:
You can only define an application when you are working in global scope.

An application 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. See the example below.

You have to take care if you define the application detection for a screen that can also have a popup window (also called child window). To define the detection rule in this case, you have to use a region on the screen that will not be overlayed by the popup window.

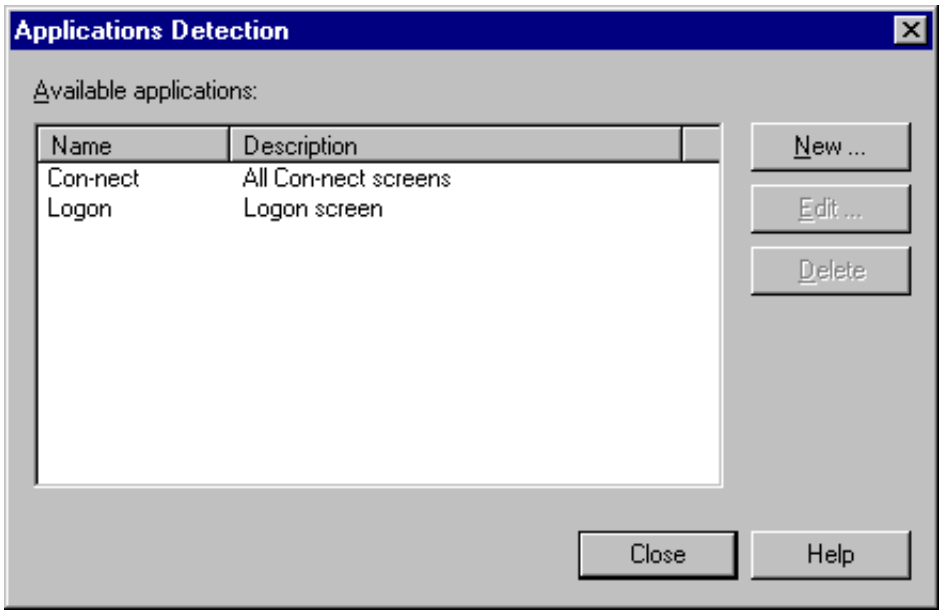
To display all defined applications

1. Make sure that you are working in global scope. Otherwise, the **Applications Detection** command is not available.

2. From the **File** menu, choose **Applications Detection**.

   Or:
   Choose the following toolbar button:

   The Applications Detection dialog box appears showing the names of all applications that have already been defined.



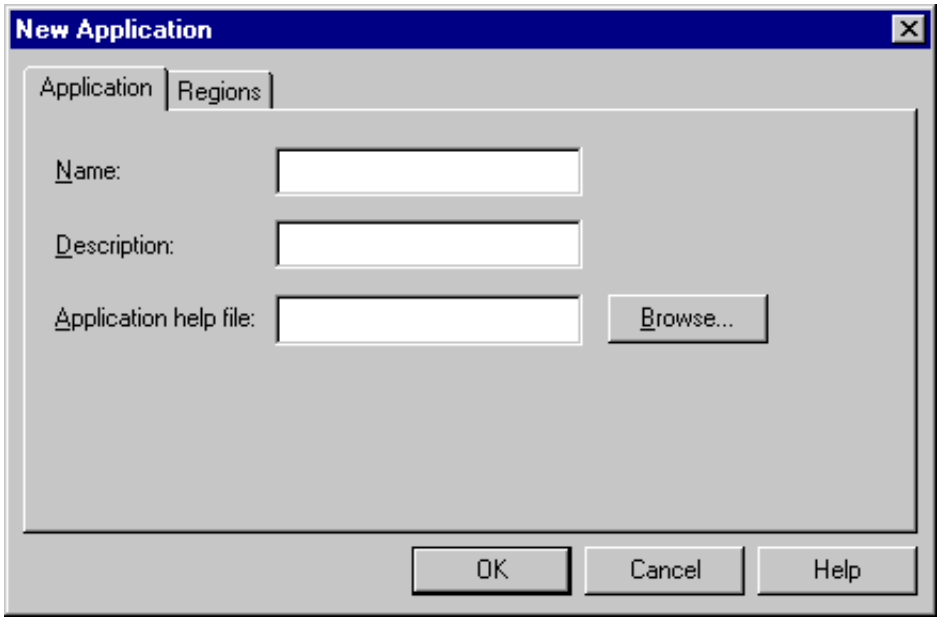
The following command buttons are provided:

New	Add a new application (see below).
Edit	Modify the selected application. Alternative: double-click an application to modify it.
Delete	Delete the selected application. You are first asked whether you want to delete the application. When you choose the **Yes** button, you are then asked whether you want to delete the corresponding folder and all the files (e.g. images, DLLs, user exits) contained in it. If you want to keep the folder, choose the **No** button; the application, however, is deleted. When you keep the folder, it is recommended that you rename it. Otherwise, an error message is shown when you later want to create an application which has the same name as the deleted application.

To add an application

1. Make sure that the screen file that is to be used as a template is shown in the scope window.
2. Display the Applications Detection dialog box as described above.
3. Choose the **New** button.

   The New Application dialog box appears.



4. In the **Name** text box, specify a name for the application.

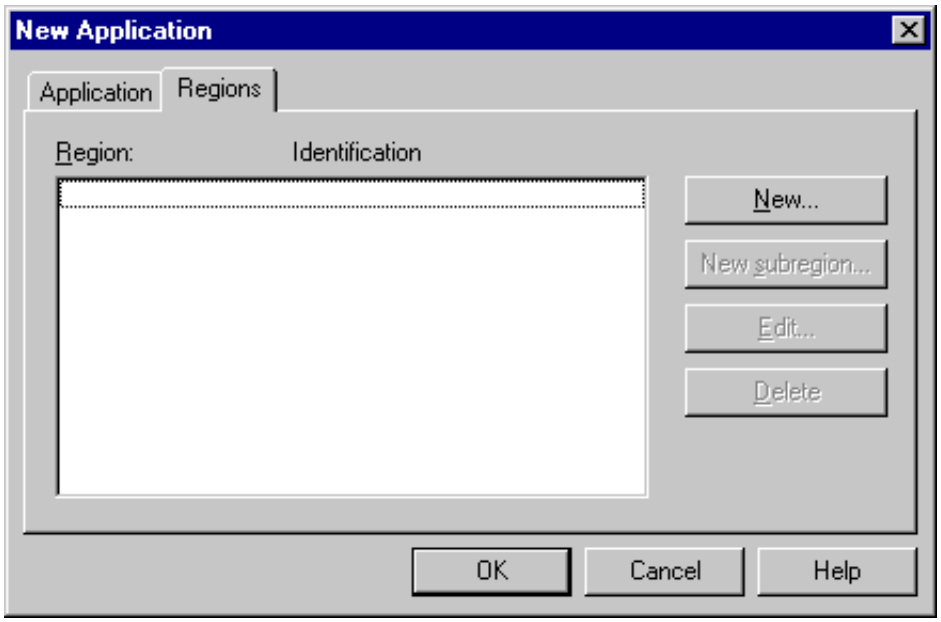
   A folder with this name will be created. All rules that you define for this application will be written to the *Defaults.ini* file in this folder. It is not possible to modify the application name once the application has been added.

5. Optionally - in the **Description** text box, specify a short description for the application.

6. Optionally - in the **Application help file** text box, specify the path to the help file that has been created for this application. Or choose the **Browse** button to select the help file from the Open dialog box.

   See *Defining a Help File for an Application* for further information.

7. Select the Regions page.



   You now have to define a region, i.e. a unique string which appears on each host screen which belongs to your application. Whenever this string is detected on a screen, the rules that have been defined for this application will be applied.

   The following command buttons are provided:

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

8. Choose the **New** button.

   Note:
   The Creating Regions dialog box providing help information is now shown if the command **Dialog "Creating Regions"** is enabled in the **View** menu.

9. Use the mouse to select a string in the global scope window.

   When you release the mouse button, the dialog box is shown again. It now contains the string you have selected and its coordinates (row and columns). A region is indicated by the letter R which is shown to the left of the coordinates.

   The string which is shown to the right of the coordinates is the text that has to be detected in the defined region of the host screen in order to identify the application.

10. If you want to define a subregion, select the region for which you want to define the subregion in the dialog box. Then choose the **New subregion** button and proceed as if defining a region.

    When you define a subregion, the application can only be detected when the region *and* all defined subregions are detected on the host screen.

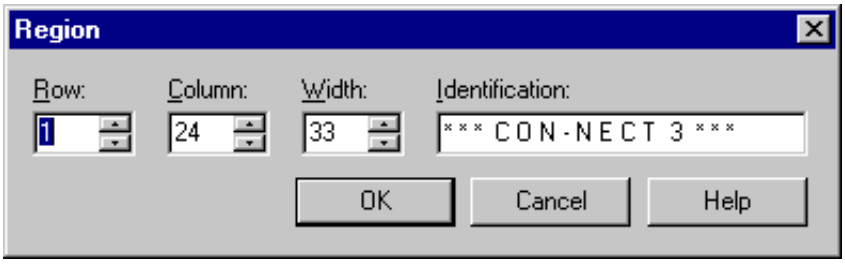
11. Choose the **OK** button to save your modifications.

    The new application is now shown in the Applications Detection dialog box.

12. Choose the **Close** button to close the Applications Detection dialog box.

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). It shows the string that is used for identification and its position in the character screen (row, column and width).



This dialog box is helpful, if the string that is used to identify a character screen does not always start in the same column of a row. For example, the string shown in the above example dialog box usually starts in column 24 and has a length of 33 characters. If there were also screens in which this string starts in different positions between column 10 and 24, you can manually adjust the values in the **Column** and **Width** spin boxes. In this case, you would set the column to 10 and the width to 46.

Example for Regions and Subregions

The application DEMO can be detected when region 1 *or* region 2 is detected, *or* when region 3 *and* all defined subregions (subregion 1 *and* subregion 2) are detected:

Region 1:	*** EMPLOYEES ***
Region 2:	*** VEHICLES ***
Region 3:	*** NAME OF EMPLOYEE ***
• Subregion 1:	*** Personal Data ***
• Subregion 2:	*** Job ***

Defining a Help File for an Application

If you have created a help file for an application, the user can invoke it when viewing the application. The help file must be located in the folder that has been defined in the New Application dialog box (see the above description) and its extension must be *hlp* or *chm*.

To invoke help for the current application

1. Connect to a session for which a help file has been defined.

   When using the SDK, you can choose **Connect** from the **Session** menu.

2. From the **Help** menu of the session (not the SDK **Help** menu), choose **Application Help**.

   When a help file is found, it is shown.

Defining the Last Rules Used

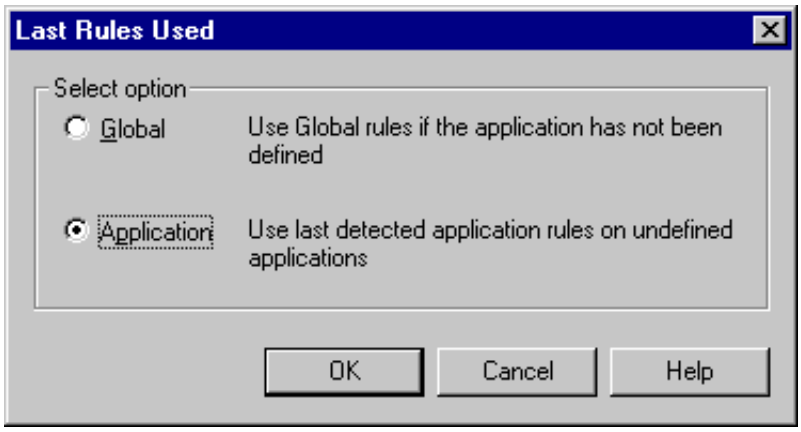
You can define the rules that are to be applied to a character screen for which an application has not been defined. By default, the rules of the last detected application are applied.

Important:
This can only be defined when you are working in global scope.

To define the rules to be applied

1. From the **Options** menu, choose **Last Rules Used**.

   The Last Rules Used dialog box appears.



2. Select one of the following option buttons:

   Global

   Global rules will be applied to a character screen for which an application has not been defined.

   Application

   The rules of the last detected application are applied to a character screen for which an application has not been defined.

3. Choose the **OK** button.

Zooming the Contents of the Scope Window

You can zoom the contents of the scope window so that you can better view the dialog controls.

To show a smaller portion in greater detail

- From the **View** menu, choose **Zoom In**.

  Or:
  Press the plus (+) key.

To show a larger portion in less detail

- From the **View** menu, choose **Zoom Out**.

  Or:
  Press the minus (-) key.

Rules Analysis

When a scope has been opened, you can display the current rules trace file in the Windows Notepad. The name of the rules trace file is *inforulesdvlp.txt*. It is located in the root of your rules repository.

Each time a host session is opened, the information in the rules trace file is overwritten.

The rules trace file logs information for each received screen: the type of scope that has been detected and the type of rule (basic, extended or multi screen) that has been applied. This information is especially helpful for developing multi screen rules.

The host session ID and environment information are written to the header section. This is followed by a block of information for each received screen.

The values next to "BDD file" and "Dialog name" indicate the type of rule that has been used:

- **Basic rule**

  BDD file: The value `L_SYS` is shown.

  Dialog name: The value `D-BAS` is shown.

- **Extended rule**

  BDD file: The name of the corresponding BDD file is shown.

  Dialog name: The name of the corresponding dialog is shown.

To display the current rules trace file

- From the **View** menu, choose **Rules Analysis**.

  The file is shown in the Windows Notepad.

Sample header section

```
 Information of rules applied by the Entire Screen Builder Server
 ================================================================

Date:        Fri Oct 17 14:49:30 2003
Mode:        Development
Repository:  C:\Program Files\Software AG\Entire Screen Builder 5\Multiscreens
Client name: PC1
Session ID:  1
```

Sample screen block where basic rules are used

```
Searching rules for the screen number 1
  Application not found. Last rules used flag: Application

  Rules for the current screen
    Application :
    Map name:
    BDD file:     ::L_SYS
    Dialog name:  ::D_BAS
```

Sample screen block where extended rules are used

```
Searching rules for the screen number 6
  Application detected
    Name:   logon
    Region: 1
  Extended dialog detected
    Region: 1
    Type:   Multi

  Rules for the current screen
    Application : logon
    Map name:
    BDD file:     tpxmain/release/tpxmain.bdd
    Dialog name:  MYPROBLEM
```

Making the Rules Available for All Users

After you have created all rules in your local development repository, you have to copy them to the production environment to make them available for all users.

To copy the rules to the production environment

1. Stop the production server.
2. Copy the entire repository folder (with all subfolders and files) to the production environment.
3. Start the production server.

See *Starting and Stopping the Entire Screen Builder Server* in Entire Screen Builder's *System Management Hub* documentation for detailed information.

Defining Basic Rules

Basic rules can be defined in global scope, application scope and map scope. Normally, they are defined in application scope.

This chapter covers the following topics:

- Overview of Tasks and Corresponding Basic Rules
- General Information on the Rules
- Using Basic Rules Together with Extended Rules
- Showing Images and Fields
- Defining Regions for Specific Rules
- Displaying Help Information when Defining Regions

See the *Basic Rules Reference* for a detailed description of each rule.

See the *First Steps* for a step-by-step description of how to apply basic rules.

Overview of Tasks and Corresponding Basic Rules

The following table gives an overview of which basic rule to use for a specific task.

To do this	**use this rule**
3D effect: apply to an image.	Image
Boxes in the character screens: convert to group boxes.	Group Box
Characters: do not display.	Delete Characters
Color: change for fields sent by the customer application.	Font and Colors for Main Dialogs Font and Colors for Child Dialogs
Constant image (always shows the content of the same file): display in the dialog.	Image
Controls: adjust size when the font is changed.	Control Attributes
Dialog background: change color or define image.	Font and Colors for Main Dialogs Font and Colors for Child Dialogs
Dialog with extended rules: place at the top, bottom, left or right of a dialog with basic rules.	Frames
Dynamic image (shows the content of a file that has the same name as the string found in the defined region of the screen): display in the dialog.	Image
Filler character used in the customer application: remove.	Control Attributes

To do this	use this rule
Font: change for fields sent by the customer application.	Font and Colors for Main Dialogs Font and Colors for Child Dialogs
Function keys: detect the location in the character screen.	Function Keys
Help for the UNIX and OpenVMS version of Natural: define the help character so that the user can invoke field help by clicking the right mouse button.	Control Attributes
Line: convert a line of a character screen to a Windows line.	Lines
Line: do not display.	Delete Lines
Map: define the string that is used to detect the screen on which the rules for map scope are to be applied.	Map Detection
Message line: move to a specific line of the dialog and change its font and/or color.	Message Line
Message line: move to the status bar.	Message Line
Popup windows in the character screens: convert to child windows.	Child Window
Prompt: do not display prompts such as "Name….:" or "Command===>".	Delete Prompt
Push buttons: create push buttons which correspond to function keys. The content of a push button can be plain text, a constant image or a dynamic image.	Buttons for Main Dialogs Buttons for Child Dialogs
Text from the character screen: move to the status bar, title bar, or do not show at all.	Item
Toolbar: display a toolbar with buttons which correspond to the function keys. The content of a toolbar button can be the function key number or an image.	Function Keys Toolbar BS2000 Keys Toolbar 3270 Keys Toolbar
User exit: change any field or data at runtime (for example, convert a specific currency to Euro).	User Exit
VGA: support standard VGA screens (640x480).	Control Attributes

General Information on the Rules

To define a basic rule, choose the corresponding command from the **Basic** menu.

The table below gives information on the default behavior of each rule, in which scope and in which type of window a specific rule can be defined, and whether it can be used together with extended rules (see also: *Using Basic Rules Together with Extended Rules*).

The table also contains some important commands from other menus. The name of the menu is then shown in parentheses.

Rule or menu command	**Default behavior when rule is not defined**	**Scope in which the rule can be defined**	**Rule can be applied in parent window**	**Rule can be applied in child window**	**Rule can be used together with extended rules**
3270 Keys Toolbar	Not applicable	Global, Application, Map	Yes	Yes	Yes
Application Detection (**File** menu)	Not applicable	Global	Yes	Yes	Not applicable
BS2000 Keys Toolbar	Default images are displayed for the detected host keys.	Global, Application, Map	Yes	Yes	Yes
Buttons for Main Dialogs	Not applicable	Global, Application, Map	Yes	No	No
Buttons for Child Dialogs	Not applicable	Global, Application, Map	No	Yes	No
Child Window	Not applicable	Global, Application	Yes	Yes	Yes
Control Attributes:					
• Adjust control size to font	Control size is automatically adjusted to font.	Global, Application, Map	Yes	Yes	Yes (except Push Button and Tab Control rules)
• Support standard VGA screens	Standard VGA screens are not supported.	Global, Application, Map	Yes	Yes	No
• Help character	A help character is not used.	Global, Application, Map	Yes	Yes	Yes (Edit Boxes rule only)

Rule or menu command	Default behavior when rule is not defined	Scope in which the rule can be defined	Rule can be applied in parent window	Rule can be applied in child window	Rule can be used together with extended rules
• Filler character	Space is used as default filler character.	Global, Application, Map	Yes	Yes	Yes (Edit Boxes rule only)
• Host key sent on double-click	ENTER	Global, Application, Map	Yes	Yes	Yes (Edit Boxes and Static Text rules only)
• Host key sent for Help	PF1	Global, Application, Map	Yes	Yes	Yes (Edit Boxes and Static Text rules only)
Delete Characters	Not applicable	Global, Application, Map	Yes	Yes	No
Delete Lines	Not applicable	Global, Application, Map	Yes	No	No
Delete Prompt	Not applicable	Global, Application, Map	Yes	Yes	Yes (List Box Controls, List View Controls and Static Text rules only)
Detection (**Extended** menu)	Not applicable	Application, Map	Yes (dialog name is taken from main window)	Yes (dialog name is taken from child window)	Not applicable
Font and Colors for Main Dialogs	Font: Courier New 8 black. Dialog background: currently defined Windows color.	Global, Application, Map	Yes	No	No
Font and Colors for Child Dialogs	Font: Courier New 8 black. Dialog background: currently defined Windows color.	Global, Application, Map	No	Yes	No

Rule or menu command	Default behavior when rule is not defined	Scope in which the rule can be defined	Rule can be applied in parent window	Rule can be applied in child window	Rule can be used together with extended rules
Frames	Not applicable	Global, Application, Map	Yes	No	No
Function Keys	Not applicable	Global, Application, Map	Yes	Yes	Yes
Function Keys Toolbar	Default images are displayed for the detected host keys.	Global, Application, Map	Yes	Yes	Yes
Group Box	Not applicable	Global, Application, Map	Yes	Yes	No
Image	Not applicable	Global, Application, Map	Yes (depending on where this rule is to be applied: parent window, child window or both)	Yes (depending on where this rule is to be applied: parent window, child window or both)	No
Item	Not applicable	Global, Application, Map	Yes (depending on where this rule is to be applied: parent window, child window or both)	Yes (depending on where this rule is to be applied: parent window, child window or both)	Window title, dialog title: Yes (only used if not defined in dialog resource). Status bar: No (the extended rules are always applied). Non-display: No.
Last Rules Used (**Options** menu)	The rules of the last detected application are applied to the character screen.	Global	Yes	Yes	Yes
Lines	Not applicable	Global, Application, Map	Yes	Yes	No

Rule or menu command	Default behavior when rule is not defined	Scope in which the rule can be defined	Rule can be applied in parent window	Rule can be applied in child window	Rule can be used together with extended rules
Map Detection	Not applicable	Application	Yes (map name is taken from main window)	Yes (map name is taken from child window)	No
Message Line	Not applicable if this rule is not defined. If this rule is defined and a font has not been defined: Courier New 8 black.	Global, Application, Map	Yes	Yes (the message line can be detected inside the child window or in the parent window)	Yes
User Exit	Not applicable	Global, Application, Map	Yes	Yes	Yes

Using Basic Rules Together with Extended Rules

When a control in a dialog that has been created with a resource editor is linked to a specific field on the character screen, specific basic rules are also applied to this control, in addition to corresponding extended rule.

The following basic rules are always applied when an extended rule is detected:

3270 Keys Toolbar
BS2000 Keys Toolbar
Child Window
Function Keys
Function Keys Toolbar
Message Line
User Exit
Last Rules Used (**Options** menu)

Several basic rules are only used when a specific extended rule is detected:

Extended rule	Basic rule(s) that will be applied
Check Boxes	The following option of the Control Attributes rule: Adjust control size to font.
Edit Boxes	The following options of the Control Attributes rule: Adjust control size to font Help character Filler character Host key sent on double-click Host key sent for Help
Group Boxes	The following option of the Control Attributes rule: Adjust control size to font.
List Box Controls (including combo box controls)	Delete Prompt and the following option of the Control Attributes rule: Adjust control size to font.
List View Controls	Delete Prompt and the following option of the Control Attributes rule: Adjust control size to font.
Push Buttons	Basic rules are not applied.
Radio Buttons	The following option of the Control Attributes rule: Adjust control size to font.
Static Text	Delete Prompt and the following options of the Control Attributes rule: Adjust control size to font Host key sent on double-click Host key sent for Help
Tab Controls	Basic rules are not applied.

The following options of the Item rule are overridden if the same information has been defined in a dialog resource (Resource Editor >Dialog >Properties >Caption):

Window title
Dialog title

Note:
The extended rules are always applied for the status bar. The corresponding setting of the Item rule is not considered.

Showing Images and Fields

These functions only apply to a screen file shown in the scope window. They do not apply to a dialog containing extended rules.

As long as the **Image** command is selected in the **Basic** menu (indicated by a checkmark), all defined images are shown in the scope window. When this command is not selected, you can also show the defined images as described below.

You can also show the fields that have been defined on the character screen.

To show all defined images

- From the **View** menu, choose **Show Images**.

  When an image has not been defined, this command is not available.

  Or:
  Choose the following toolbar button:

To show all fields in the character screen

- From the **View** menu, choose **Show Fields**.

  Or:
  Choose the following toolbar button:

Defining Regions for Specific Rules

The commands in the **Edit** menu are only available when you are defining a basic rule that requires the definition of a region (i.e. when the dialog box showing, for example, all Image or Map Detection rules is shown). Instead of using the commands from the **Edit** menu, you can also choose the corresponding command button (**New** or **Delete**) in the dialog box.

These commands do not apply to a dialog containing extended rules.

See the *Basic Rules Reference* for information on how to define a region for a specific rule.

To define a new region

1. From the **Edit** menu, choose **New Region**.

   Or:
   Choose the following toolbar button:

2. In the scope window, draw an outline around the desired region.

   A new rule is automatically created and the corresponding dialog box is shown.

To select a defined region

1. From the **Edit** menu, choose **Select Region**.

   This command is active by default (i.e. you just have to click the region in order to select it).

   Or:
   Choose the following toolbar button:

2. In the scope window, click the desired region.

   The corresponding definition is automatically highlighted in the dialog box that has been displayed for the selected rule.

   Instead of selecting a region as described above, you can simply select the entry in the dialog box that has been displayed for the selected rule. This also selects the corresponding region in the scope window.

To delete the selected region

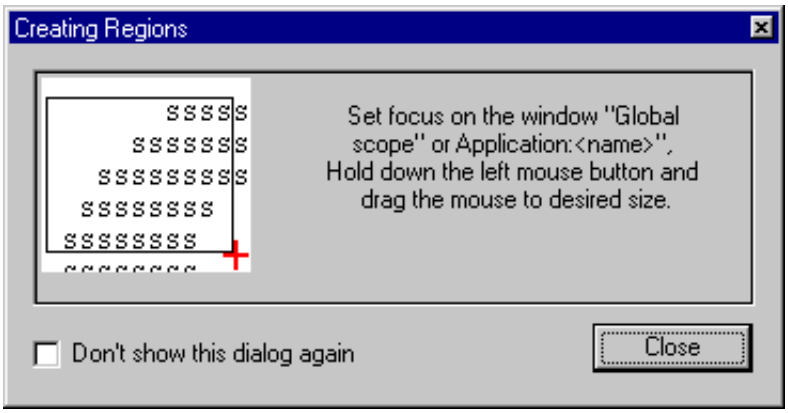
- From the **Edit** menu, choose **Delete**.

  Or:
  Press DEL.

  Or:
  Choose the **Delete** button in the dialog that has been displayed for the selected rule.

Displaying Help Information when Defining Regions

When you define a region for a rule or select a field, a dialog box may appear containing help information. Whether the dialog box appears depends on your settings in the **View** menu.



When you select the **Don't show this dialog again** check box, this dialog box will not shown be shown the next time, and the command **Dialog "Creating Regions"** in the **View** menu is disabled.

To switch help information display on and off

- From the **View** menu, choose **Dialog "Creating Regions"**.

  When help information will be shown, a check mark is shown next to this menu command.

Defining Extended Rules

Extended rules apply to GUI controls such as push buttons, radio buttons or check boxes. These GUI controls are defined using a resource editor.

This chapter covers the following topics:

- Using a Resource Editor
- Opening a DLL
- Opening a Dialog
- Closing a Dialog
- Defining the Dialog Properties
- Defining the Control Properties
- Defining the Font
- Using Styles
- Saving the DDT File
- Building the BDD File
- Detecting the Screen on which the Extended Rules are to be Applied (Single Screen)

See the *Extended Rules Reference* for a detailed description of each rule.

See the *First Steps* for a step-by-step description of how to define extended rules.

See also: *Using Basic Rules Together with Extended Rules*.

Using a Resource Editor

You can use any resource editor that can create standard Windows resource DLLs (dynamic link libraries), for example Microsoft Visual Studio. Other editors, such as Visual Basic 6.0, use a different format for resource DLLs and can therefore not be used with Entire Screen Builder.

The following topics are covered below:

- Creating a DLL
- Reserved Control IDs
- Reserved Dialog Names

Creating a DLL

In the resource editor, a DLL has to be created containing a dialog for each screen that is to be presented in an extended dialog. You can only use the controls that are supported by Entire Screen Builder. See the *Extended Rules Reference* for further information.

The following description applies when you use Microsoft Visual C++ Version 6, which is part of Microsoft Visual Studio, to create the DLL.

To create a DLL containing a dialog using Microsoft Visual C++

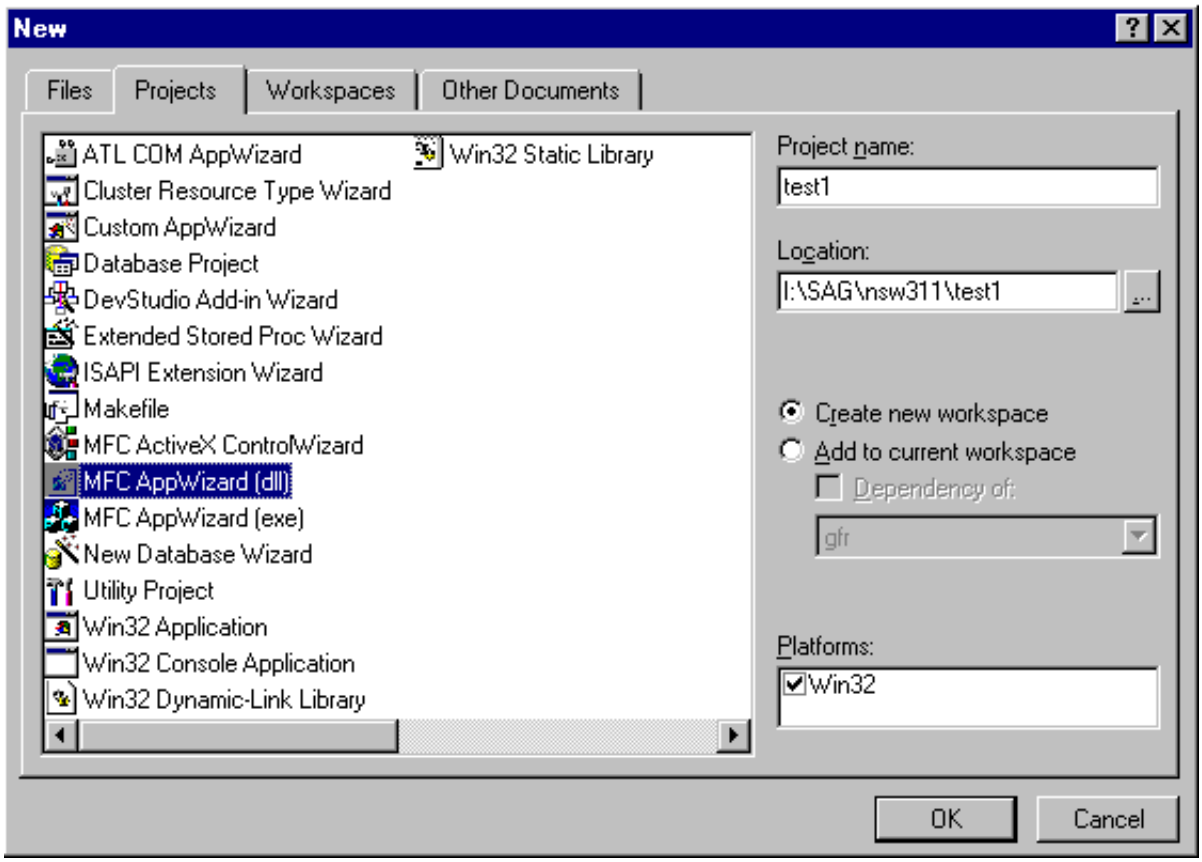
1. From the **File** menu, choose **New**.

   The New dialog box appears.

2. Make sure that the Projects page is shown.

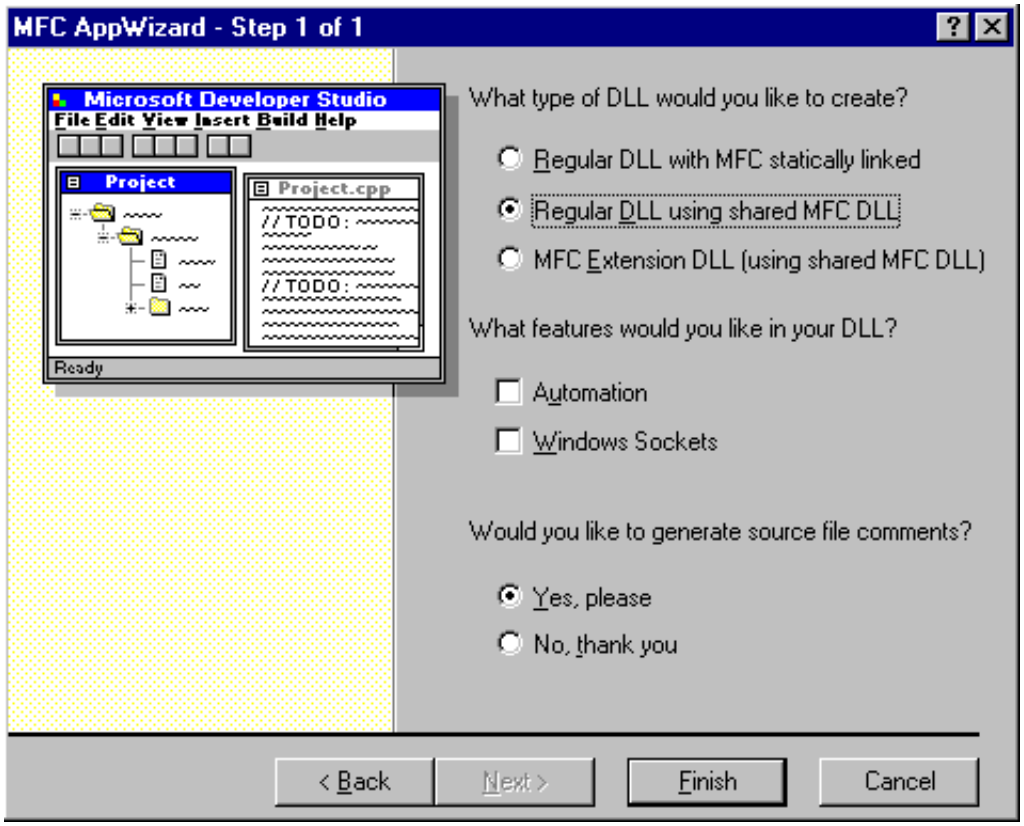
3. Select **MFC AppWizard (dll)**.

4. In the **Project name** text box, specify the desired project name.



5. Choose the **OK** button.

   The following dialog box appears.



Leave the default options as shown in the above dialog box.

6. Choose the **Finish** button.

   The New Project Information dialog box appears.

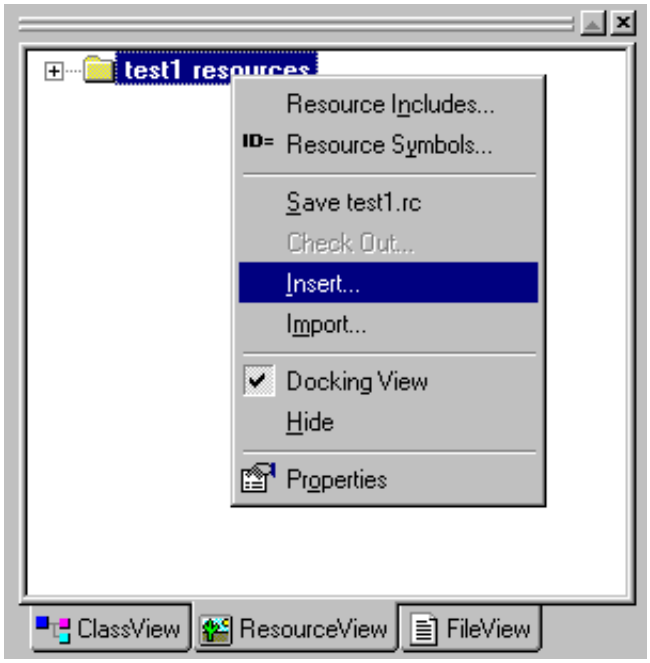
7. Choose the **OK** button.

   The DLL is now created. The name you have defined for the project is shown in the workspace window.

8. Make sure that the ResourceView page is shown.

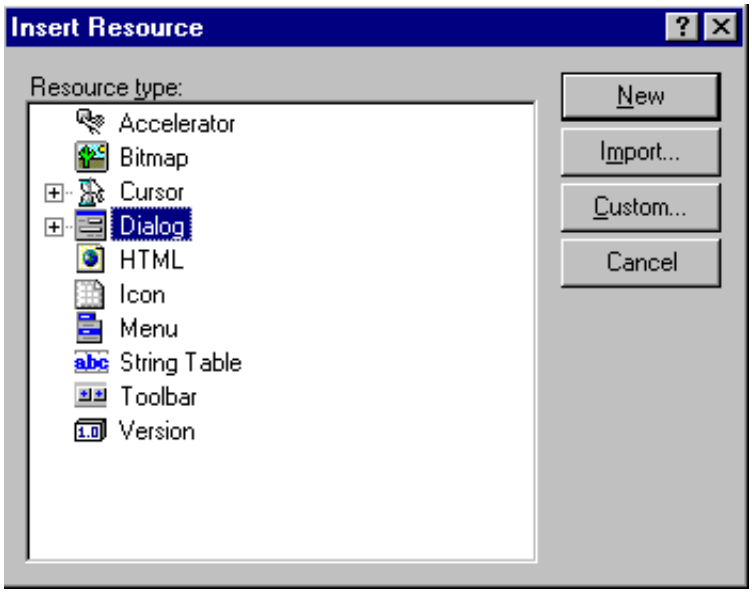
9. Select the resource name and invoke the context menu.

10. From the context menu, choose **Insert**.



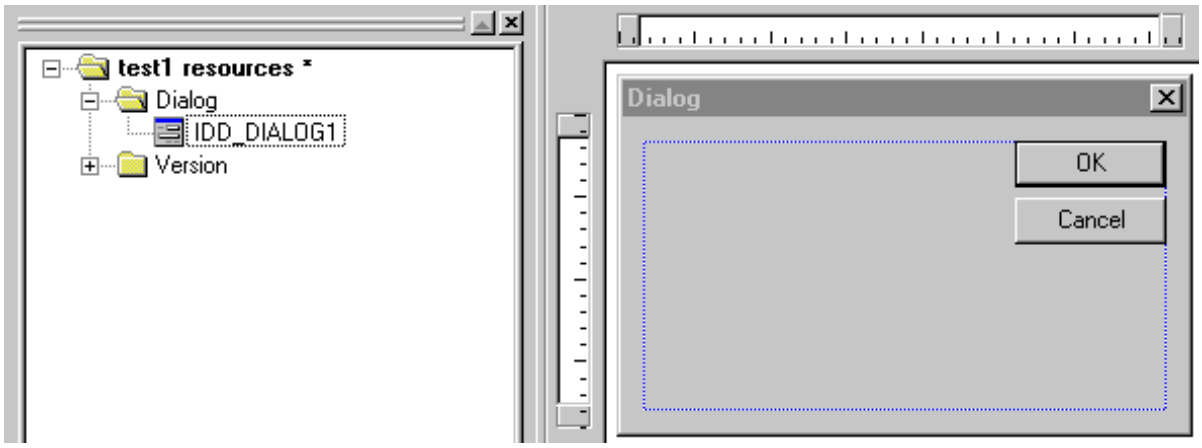
The Insert Resource dialog box appears.

11. Select the resource type **Dialog**.



12. Choose the **New** button.

    A new dialog is now shown. Its name is shown in the tree.



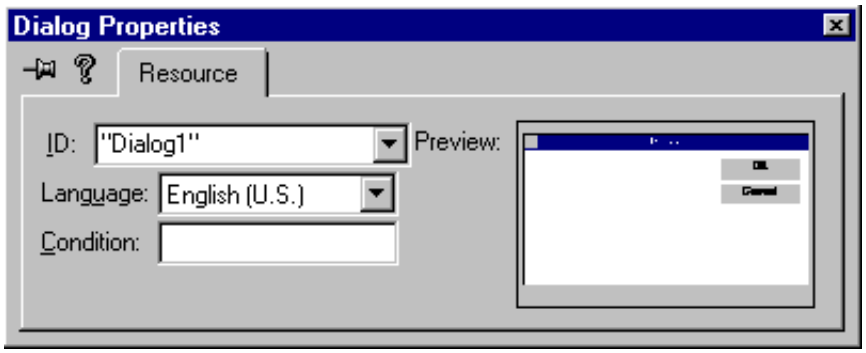
Important:
Change the ID of the new dialog to a string value as described below.

13. In the tree, select the dialog name and invoke a context menu.

14. From the context menu, choose **Properties**.

    The Dialog Properties dialog box appears.

15. In the **ID** text box, specify a new ID enclosed in quotation marks.



16. Press ENTER to close the dialog box.

    You can now design the dialog as follows:

    - either design a new dialog, or
    - load an existing screen file that has been saved using the Entire Screen Builder SDK into the dialog (see the description below).

    When the dialog design has been completed, you have to build the DLL.

17. From the **Build** menu, choose **Build *name*.dll**.

    Or:
    Press F7.

When the DLL has been built successfully, you can open it using the Entire Screen Builder SDK. See *Opening a DLL* for further information.

To load an Entire Screen Builder screen file into the dialog

1. From the **File** menu, choose **Open**.

   The Open dialog box appears.

2. Select the desired screen file.

3. Choose the **Open** button.

   Information similar to the following is now shown:

4. Open the dialog (for example, by double-clicking its name).

   The content of the screen file is now shown in a dialog.

5. Make sure that this dialog is selected.

6. From the **Edit** menu, choose **Select All**.

   Or:
   Press CTRL+A.

7. From the **Edit** menu, choose **Copy**.

   Or:
   Press CTRL+C.

8. In the tree view, open the dialog you have created in the previous exercise.

9. If required, resize the dialog.

10. From the **Edit** menu, choose **Paste** to copy the contents of the screen file to the dialog.

    Or:
    Press CTRL+V.

Reserved Control IDs

The following control IDs are reserved for Entire Screen Builder. Therefore, all controls that you create in the resource editor (for example, push buttons, group boxes or list boxes) must to have a control ID bigger than 0x5000 (20480).

Control ID	Description
0x0000 - 0x0009	Reserved for Windows
0x000A - 0x0100	Undefined
0x0101 - 0x2480	Screen ID (see *Field Identification* for further information)
0x2481 - 0x24FD	Undefined
0x24FE	Message line (basic rules)
0x24FF	ENTER key
0x2500 - 0x2563	Host keys
0x2564 - 0x25C7	Images
0x25C8	Group box
0x25C9 - 0x25CF	3270 special keys
0x25D0 - 0x26D0	Popup menu
0x26D1 - 0x2AF7	Undefined
0x2AF8	Message line (extended rules)
0x2AF9 - 0x4999	Undefined

Reserved Dialog Names

The following dialog names are reserved for Entire Screen Builder. They are used by the Frames rule.

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

Important:
The quotation marks are part of the frame name.

Opening a DLL

A DLL contains the dialogs that have been created using a resource editor.

Important:
You can only open a DLL when you are working in application or map scope.

If you want to open another DLL, you must first close an already open dialog.

To open a DLL

1. From the **File** menu, choose **Open DLL**.

   Or:
   Choose the following toolbar button:

   The Open DLL dialog box appears.

2. Select the desired DLL.

3. Choose the **Open** button.

   A dialog box appears providing for selection all dialogs that have been defined in the DLL (see below).

Opening a Dialog

After the DLL has been opened, you can open a dialog contained in it. To do so, you can either use a menu command or the drop-down list box in the toolbar. Only one dialog can be shown at one time. If you want to display another dialog, you must first close an already open dialog.

To open a dialog using a menu command

1. From the **Extended** menu, choose **Open Dialog**.

   Or:
   Choose the following toolbar button:

   A dialog box appears listing all dialogs that have been defined in the DLL.

2. Select the desired dialog.
3. Choose the **OK** button.

To open a dialog using the drop-down list box in the toolbar

- From the drop-down list box in the toolbar, select the desired dialog.

Closing a Dialog

You can either close a dialog with the standard button in the upper right corner of the dialog or with a command.

To close a dialog using a command

- From the **Extended** menu, choose **Close**.

  Or:
  Choose **Close** from the context menu that is invoked by pressing the right mouse button.

Defining the Dialog Properties

You can define a background color or image for the current dialog. When a dialog style has been defined for the background color, you also define this style.

Note:
You can also use the font toolbar to define a backgound color or style.

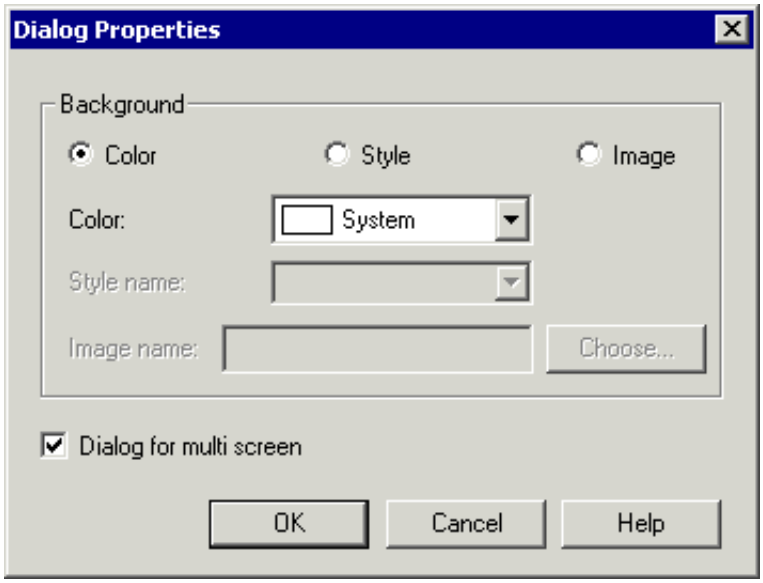
To define the dialog properties

1. Open the dialog for which you want to define the dialog properties.

2. From the **Extended** menu, choose **Dialog Properties**.

   Or:
   Click the right mouse button in an area of the dialog that does not contain a control and choose **Dialog Properties** from the resulting context menu.

   The Dialog Properties dialog box appears.



3. In the Background group box, select one of the following option buttons:

   Color

   When this option button is selected, the **Color** drop-down list box is available and you can select the desired color from this drop-down list box.

   When you select **System** from this drop-down list box, the background color as defined under Windows will be used.

   When you select **Custom...** from this 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...**.

Style

When this option button is selected, the **Style name** drop-down list box is available. When a style has been defined, you can select it from this drop-down list box. See *Defining a Dialog Style*.

Image

When this option button is selected, the **Image name** text box is available and you can 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*.

4. If you want to use the multi screen feature, select the following check box:

Dialog for multi screen

When this check box is selected, an additional page, *Screen Number*, is available in the control properties. See also: *Using the Multi Screen Feature*.

Note:
This check box is disabled when a tab control has been defined in the extended dialog.

5. Choose the **OK** button.

Defining the Control Properties

The context menu that appears when you click the right mouse button shows all commands that apply to the selected control. In order to apply extended rules to the selected control, you have to choose the command **Control Properties**. This command is available in the context menu and in the **Extended** menu.

To define the control properties

1. Select the control to which you want to apply extended rules.
2. From the **Extended** menu, choose **Control Properties**.

   Or:
   Invoke the context menu and choose **Control Properties**.

   Or:
   Double-click the control.

   Depending on the control that is currently selected, different pages are available in the resulting dialog box. The pages that are used by more than one control are explained in the following topics below:

   - Field Identification
   - Display Styles
   - Action
   - Mouse
   - Screen Number

   For information on the rule-specific pages not covered below, see the descriptions of the corresponding rules in the *Extended Rules Reference*.
3. Specify all required information on the different pages.
4. Choose the **OK** button.

Field Identification

In order to transfer the values that a user enters in an extended dialog to the character screen and vice versa, you have to define the corresponding field for a number of dialog controls. Each field has a unique ID based on its position in the screen. The ID is calculated as follows:

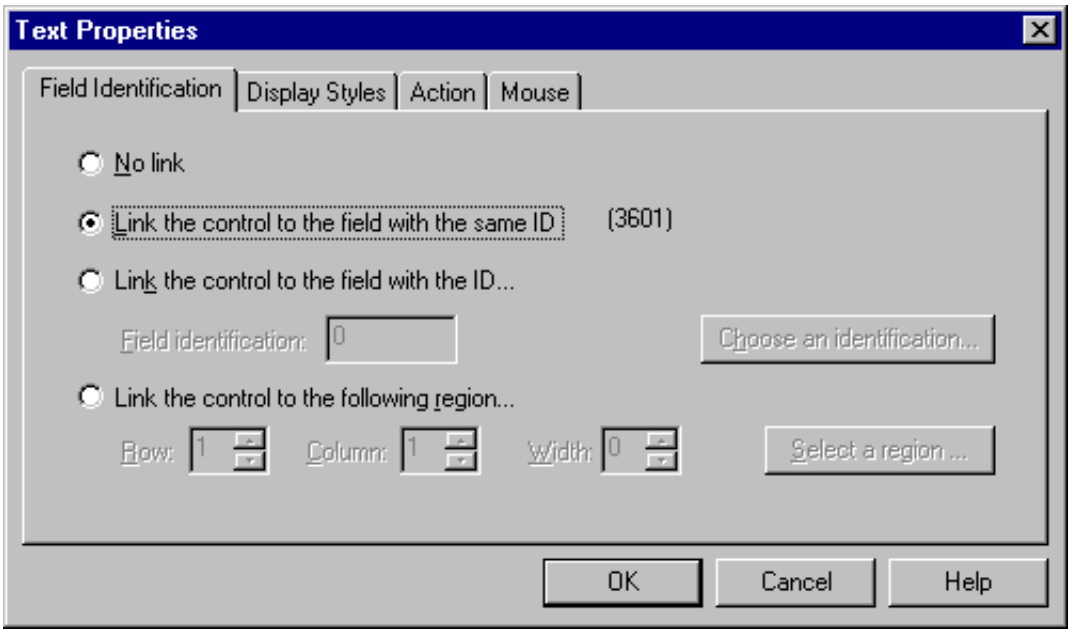
```
ID = (row * 256) + column
```

There are two different ways of creating a dialog:

- You save a character screen as a screen file and then modify it using the resource editor. In this case, it is not required that you define the corresponding fields. By saving the screen, all screen IDs (0x0101-0x2480) are automatically defined. However, if you create new dialog controls, you have to define all corresponding fields manually. By default, the option button **Link the control to the field with the same ID** (see below) is enabled for controls such as edit boxes. For a static text control, the option button **No link** is enabled by default.

- You start from scratch, i.e. you create a new dialog for a screen using the resource editor. In this case, you have to define all corresponding fields manually.

When the definition of the corresponding field is required, the Field Identification page is available.



No link

The content of the control is not changed.

For example, if you have defined a static text control with the label "None:", this label is always shown.

Link the control to the field with the same ID

The content of the control is substituted with the content of a field that has the same ID.

For example, you have defined a static text control with ID 257 and the label "None:". When the following has been defined for the field, the label "Same:" is then shown on the control:

```
ID = 257, row=1, column=1, text="Same:"
```

Link the control to the field with the ID

The content of the control is substituted with the content of a field that has another ID. In this case, you can either type the field ID in the text box or choose the **Choose an identification** button to select the desired field in the scope window.

For example, you have defined a static text control with ID 257 and the label "None:". When the following has been defined for the field, the label "Other:" is then shown on the control:

```
ID = 2580, row=10, column=20, text="Other:"
```

Link the control to the following region

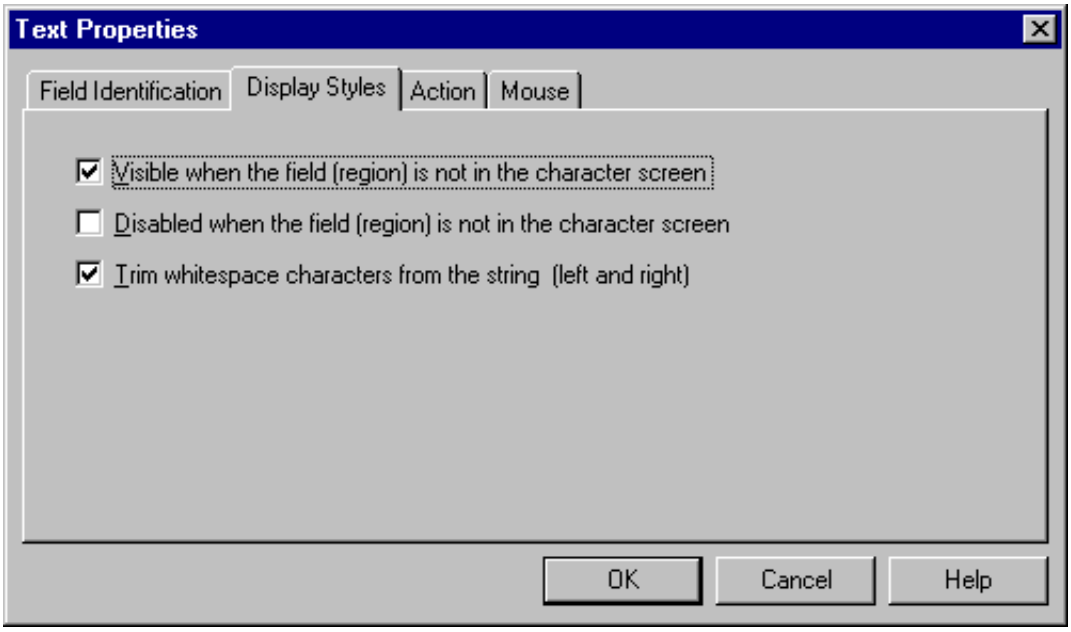
The content of the control is substituted with the content of the defined region. In this case, you can either specify the coordinates of the region using the spin boxes (row, column and width) or choose the **Select a region** button to define the desired region in the scope window.

For example, you have defined a static text control with ID 257 and the label "None:". When the following has been defined for the region, the label "Region" is then shown on the control:

```
row=20, column=5, width=10, text="Region"
```

Display Styles

When the definition of the corresponding field is required, the Display Styles page is also available (in addition to the Field Identification page). The options on the Display Styles page are only available when an option other than **No link** is selected on the Field Identification page.



Visible when the field (region) is not in the character screen

When this check box is selected, the control will be shown even if the corresponding field or region is not available in the character screen.

Disabled when the field (region) is not in the character screen

This check box is only available when the check box **Visible when the field (region) is not in the character screen** has also been selected.

When this check box is selected, the control will be disabled if the corresponding field or region is not available in the character screen. Otherwise, it is not shown at all.

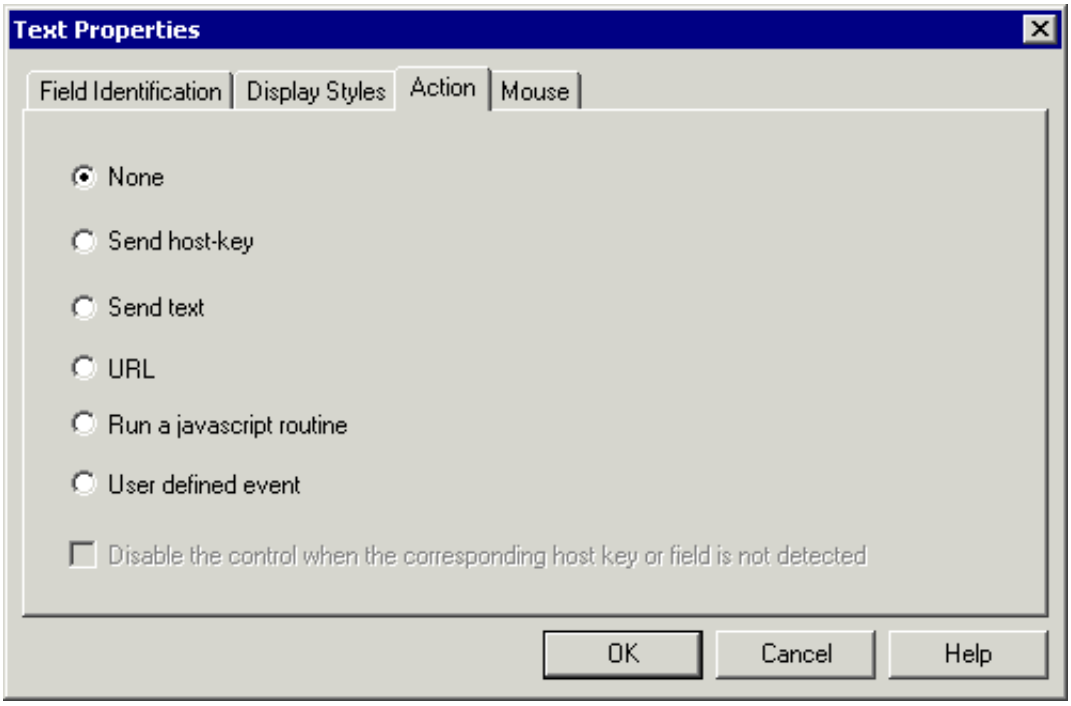
When the corresponding field for a control is a protected field (this may happen when the application changes an unprotected field to a protected field), the control is automatically disabled. The setting of this option is ignored in this case.

Trim whitespace characters from the string (left and right)

This check box is only available for static text. When this check box is selected, any blank characters before and after the string which makes up the static text are removed.

Action

The Action page is available for push button and static text controls. When you define an action, it is recommended that you enable the **Enable the single click** check box on the Mouse page.

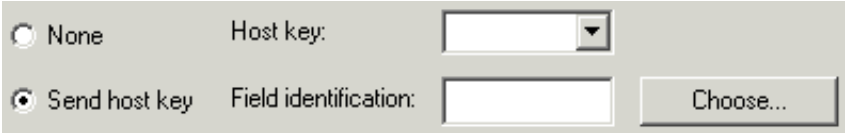


None

When this option button is selected, no action will be taken when the user chooses the control. Use this option for a push button control, if you want to display an image only.

Send host key

When this option button is selected, additional options are shown:



You can send the defined function key to the application.

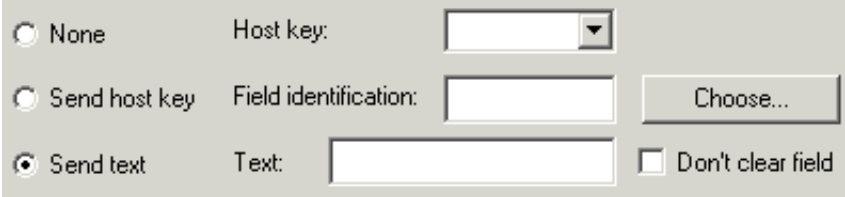
- **Host key**
  From the drop-down list box, select the host key you want to associate with the control.

  This drop-down list box also contains BS2000 keys.

- **Field identification**
  Specify the identification of the input field which receives the focus when the user chooses the control. To do so, make sure that the corresponding screen file is shown in the scope window and choose the **Choose** button. You can now click the desired field in the scope window.

Send text

When this option button is selected, additional options are shown:



You can display text in an input field and then send the defined function key to the application.

- **Host key**
  From the drop-down list box, select the host key you want to associate with the control.

  This drop-down list box also contains BS2000 keys.

- **Field identification**
  Specify the identification of the input field which receives the focus when the user chooses the control. To do so, make sure that the corresponding screen file is shown in the scope window, choose the **Choose** button and click the desired field in the scope window.

- **Text**
  In the text box, specify the text that is to be shown in the input field.

- **Don’t clear field**
  When this check box is not selected, the content of the input field is cleared before new information is written to it. However, when this check box is selected, the new information just overwrites (part of) the information in this input field.

  For example: the input field contains the text "ABCDEF" and the new text "?" is sent to this input field. When this check box is not selected, "?" is then shown in the input field. When this check box is selected, "?BCDEF" is shown in the input field.

URL

When this option button is selected, an additional text box is shown:

You can open a new browser window.

- **Address**
  Specify the URL of the web site that is to be opened in a new browser window when the user chooses the control. For example:

  http://www.softwareag.com/corporat/default.htm

Run a JavaScript routine

When this option button is selected, an additional text box is shown:

The drop-down list box provides for selection the script files stored in the *Production* folder. Select the script file that is to be executed when the user chooses the control.

See also: *General Information on Script Files* in the *Script Files* documentation.

User defined event

When this option button is selected, an additional text box is shown:

Specify the event that is to be fired when the user chooses the control:

- **Text is not specified**
  When only the option button has been selected and a text has not been defined in the text box, the event `UserDefinedEvent` will be fired into the HTML page. This enables you to have local processing in the HTML page on the client workstation using a scripting language supported by the browser (for example, VBScript).

- **Text is specified**
  Optional. You can specify a text for the user-defined event. In this case, the event `UserDefinedEvent(szText)`, including the text that you have specified, will be fired into the HTML page. This is helpful when you have specified a user-defined event for more than one control in the dialog. The event will then be identified by its name.

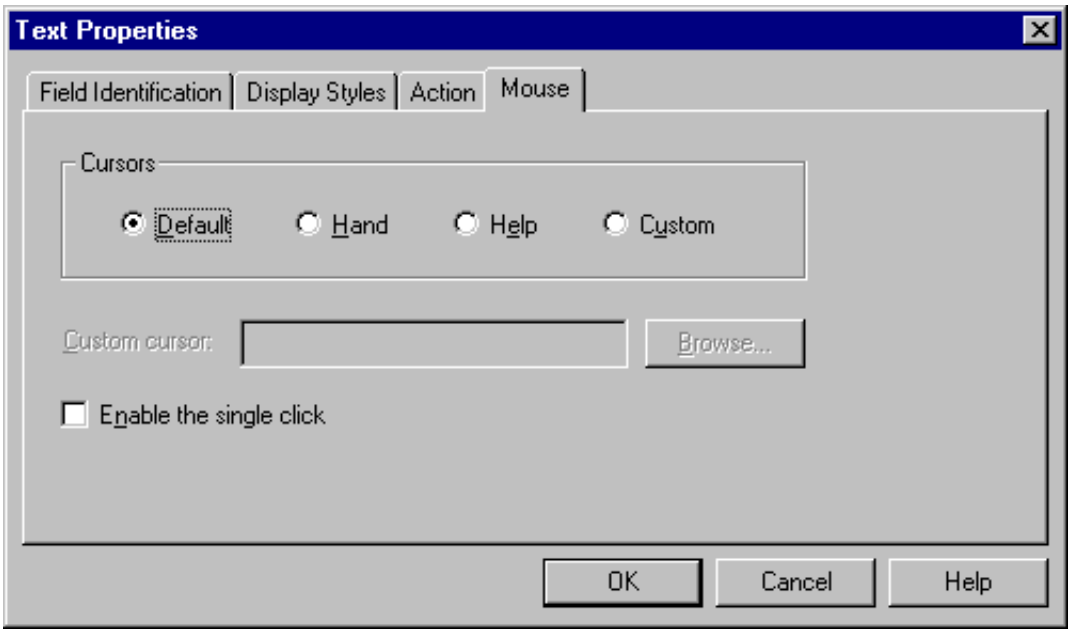
See also: *Events* in the *Web Viewer Client Functions (API)* section of the *User Exits* documentation.

Disable the control when the corresponding host key or field is not detected

This check box is only available for push buttons. It is available when an option other than **None** has been selected. When this check box is selected, the control is disabled if the corresponding host key or field is not available in the character screen.

Mouse

The Mouse page is available for most resource editor controls that are supported by Entire Screen Builder.



Cursors

You can define the type of cursor that is to be used. Select one of the following option buttons:

- **Default**
  The cursor as defined under Windows.
- **Hand**
  Hand-shape cursor.
- **Help**
  Cursor in the shape of a question mark.
- **Custom**
  You can define your own cursor (see below).

Custom cursor

When the **Custom** option button has been selected, you must specify a path and file name in this 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.

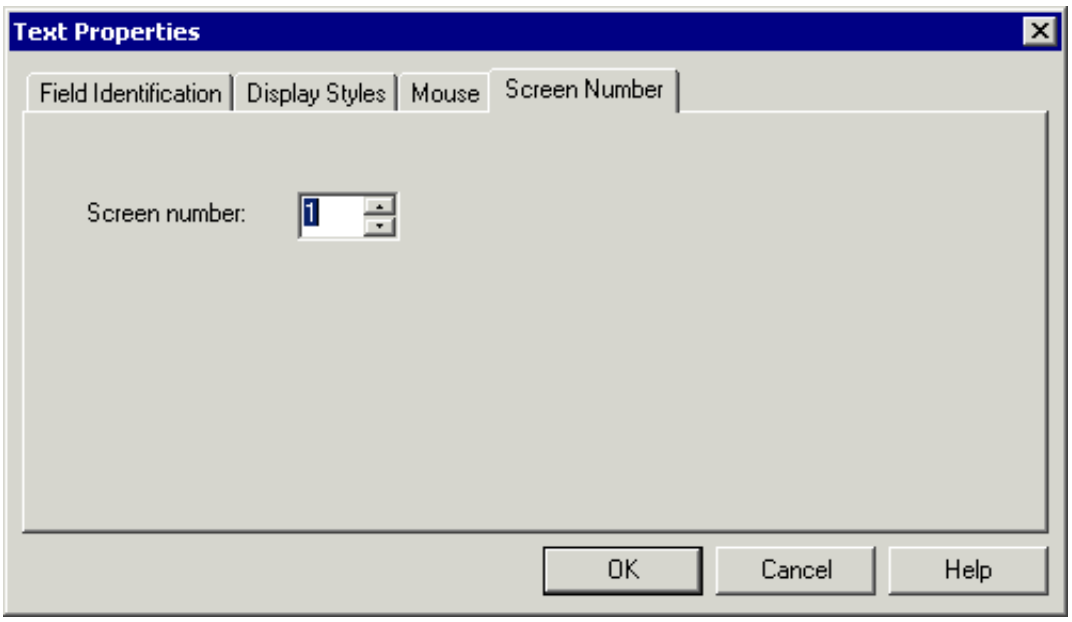
Enable the single click

When this check box is selected, a single click on a control has the same effect as a double-click. If an action has been defined for the control (**Send host key**, **Send text**, **URL** or **Run a JavaScript routine**), this action is executed after a single click. If no action has been defined, the corresponding host key that has been defined with the basic rule Control Attributes is sent to the host.

Screen Number

The Screen Number page is only available when the option **Dialog for multi screen** has been activated in the dialog properties (see *Defining the Dialog Properties*).

See also: *Using the Multi Screen Feature*.



Screen number

With the multi screen feature, fields from different terminal screens can be shown in one dialog.

You have to specify the number of the terminal screen which contains the corresponding field for the current control.

Defining the Font

You can define another font or character set for a control.

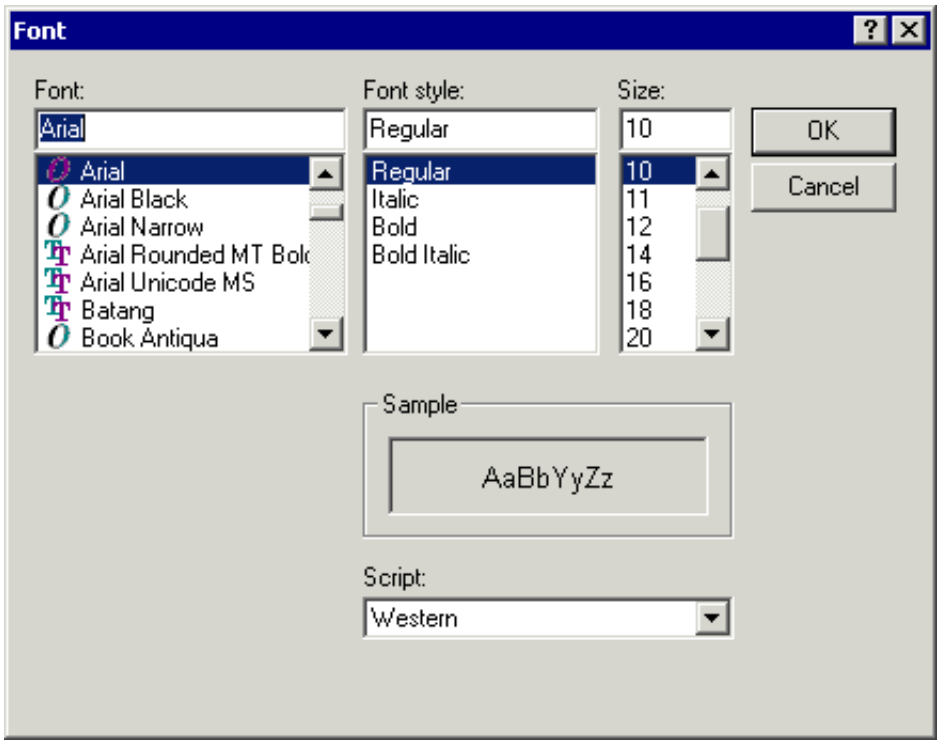
Note:
You can also use the font toolbar to define a font.

To define another font

1. Select the control for which you want to define another font.
2. From the **Extended** menu, choose **Font**.

   Or:
   Invoke the context menu and choose **Font**.

   The Font dialog box appears.



3. Select the desired font, font style and size.
4. If you want to use another character set, select it from the **Script** drop-down list box.
5. Choose the **OK** button.

Using Styles

You can define styles for frequently-used font and color settings. You can then apply these styles to your controls and dialogs.

The following topics are covered below:

- Defining a Control Style
- Defining a Dialog Style
- Applying a Style to a Control or Dialog

Defining a Control Style

Control styles can be applied to the following controls:

Check boxes
Edit boxes (one input field and several input fields)
Group boxes
List box controls (list boxes and combo boxes)
List view controls
Radio buttons
Static text

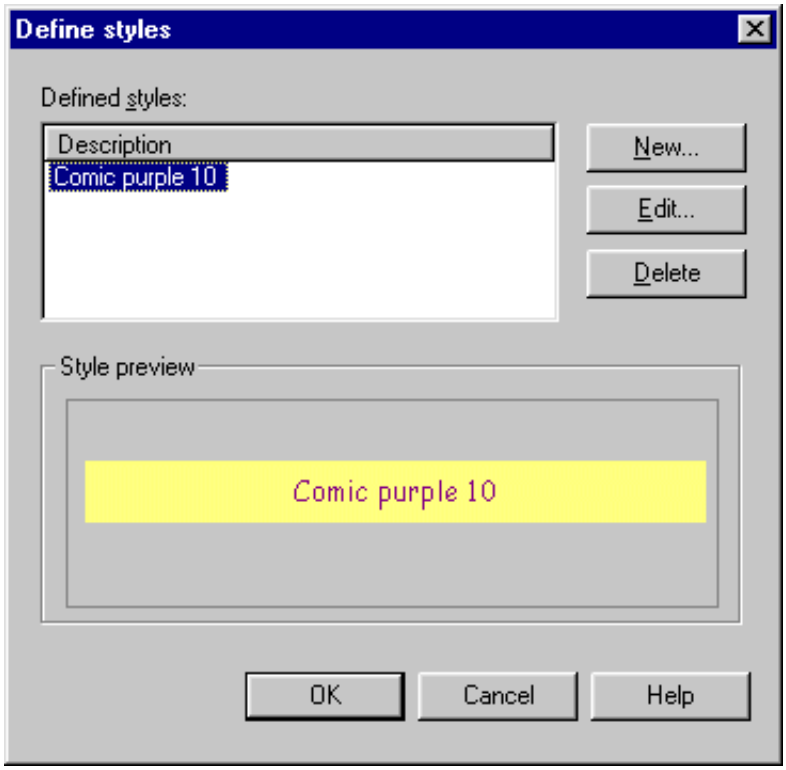
You can only define control styles when you are working in application or map scope. A dialog must not be open.

Using the Control Attributes rule, you can define that the size of a control is automatically increased or reduced according to the font that has been defined for the text in this control.

To define a new control style

1. From the **Extended** menu, choose **Define Styles** > **Control Styles**.

   The Define Styles dialog box appears.



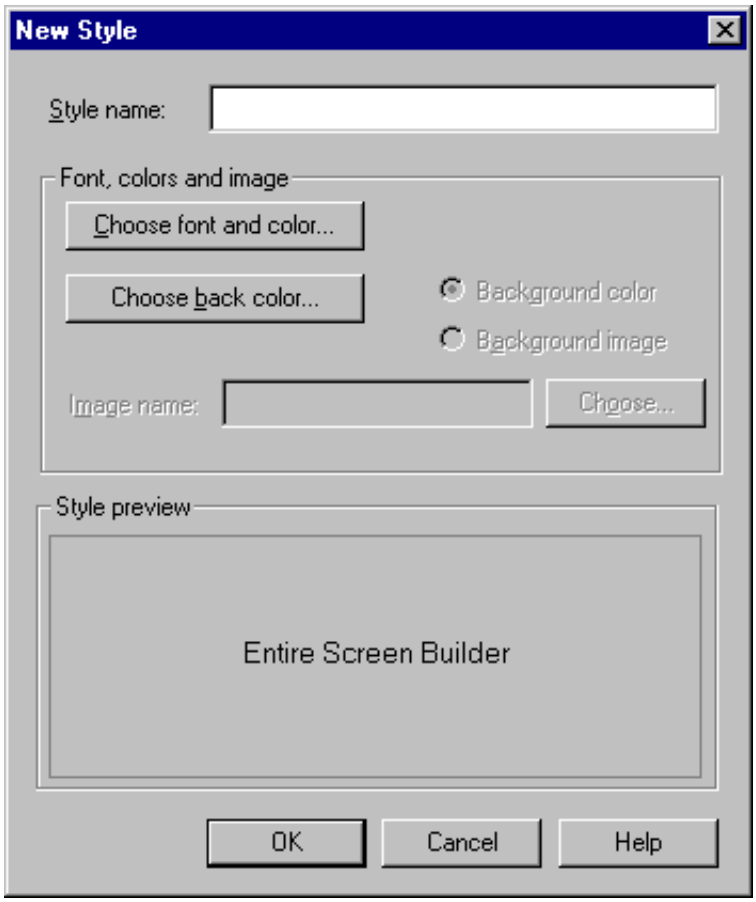
   When control styles have already been defined, the names you defined for them are shown in this dialog box. When you select a control style, the preview section shows how this style looks like.

   The following command buttons are available:

New	Define a new style (see below).
Edit	Modify the selected style.
Delete	Delete the selected style.

2. To define a new control style, choose the **New** button.

   The New Style dialog box appears.

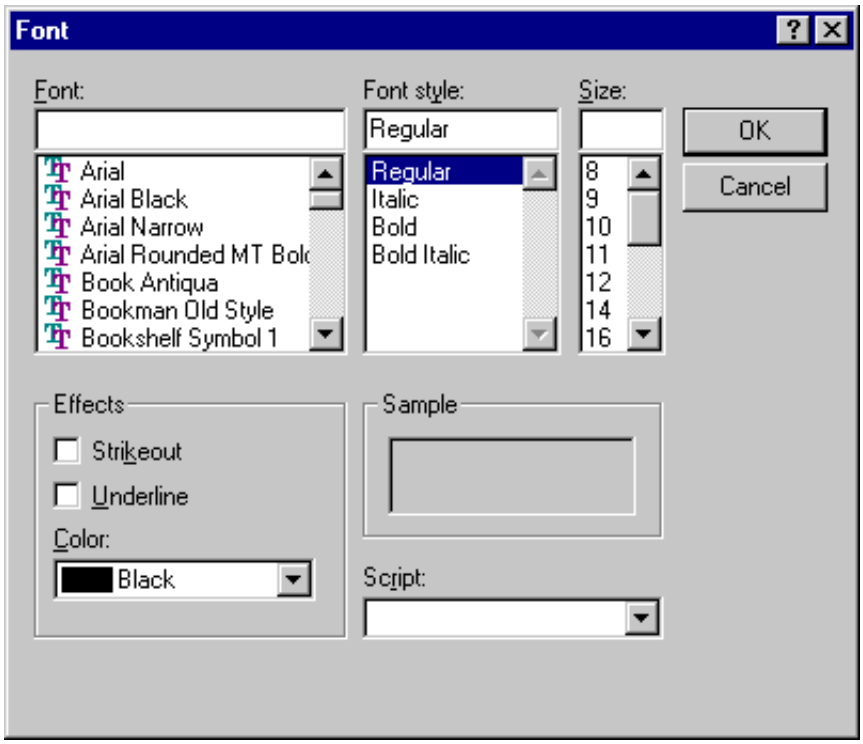


Note:
The option buttons for background color and background image are not available for control styles.

3. In the **Style name** text box, enter a name for the style.

4. To define the font and/or color, choose the corresponding command button.

   The Font dialog box appears.



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

   The new font and color are now shown in the preview section of the New Style dialog box.

6. To define the background color, choose the corresponding command button.

   The Color dialog box appears.

7. Define the desired color and choose the **OK** button.

   The new background color is now shown in the preview section of the New Style dialog box.

8. Choose the **OK** button to close the New Style dialog box.

Defining a Dialog Style

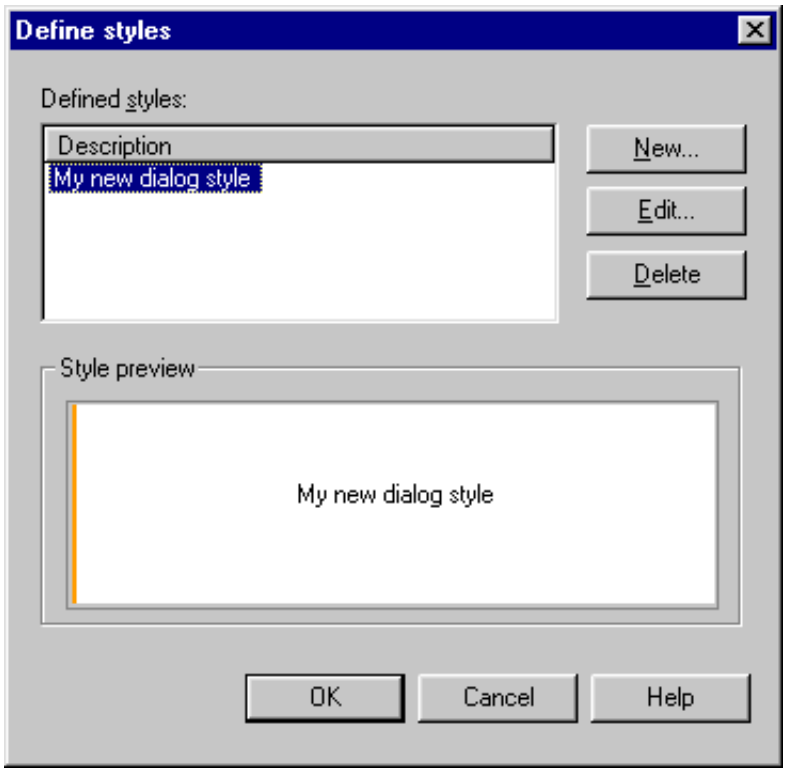
Dialog styles can only be applied to dialogs. You can define a background color or a background image.

You can only define dialog styles when you are working in application or map scope. A dialog must not be open.

To define a new dialog style

1. From the **Extended** menu, choose **Define Styles** > **Dialog Styles**.

   The Define Styles dialog box appears.



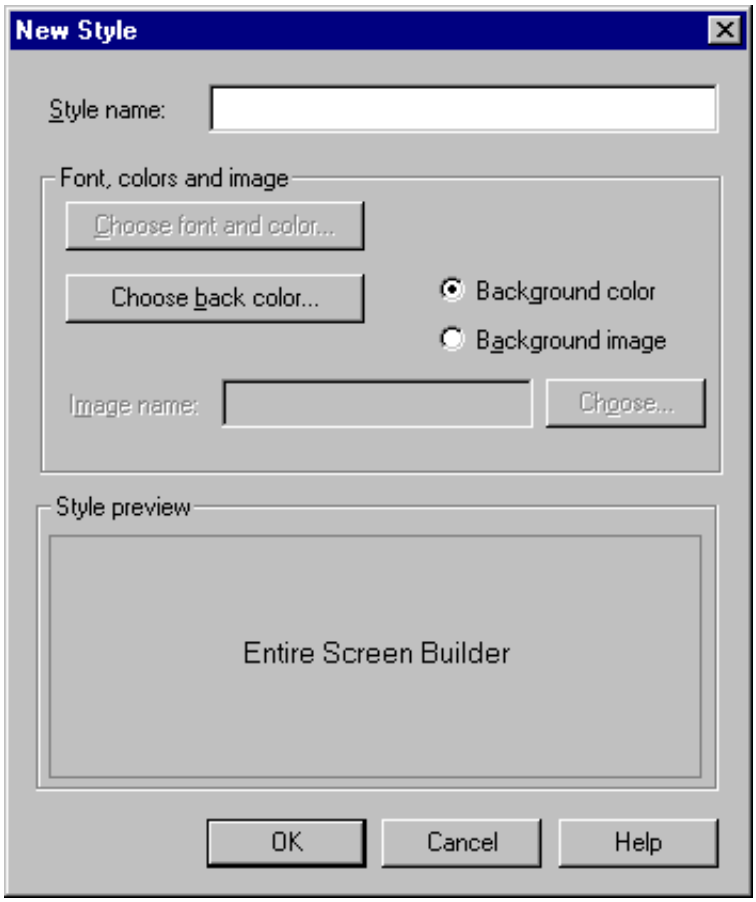
   When dialog styles have already been defined, the names you defined for them are shown in this dialog box. When you select a dialog style, the preview section shows how this style looks like.

   The following command buttons are available:

New	Define a new style (see below).
Edit	Modify the selected style.
Delete	Delete the selected style.

2. To define a new dialog style, choose the **New** button.

   The New Style dialog box appears.



Note:
The command button for choosing a font and color is not available for dialog styles.

3. In the **Style name** text box, enter a name for the style.

4. To define a background color, make sure that the **Backgound color** option button is selected and choose the **Choose back color** command button.

   The Color dialog box appears and you can define the desired color.

Or:
To define a background image, make sure that the **Backgound image** option button is selected. In the **Image name** text box, specify the path to the 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*.

When you have defined a new background color or background image, it is shown in the preview section of the New Style dialog box.

5. Choose the **OK** button to close the New Style dialog box.

Applying a Style to a Control or Dialog

When you have defined a control or dialog style, you can apply it to a control or dialog.

Note:
A dialog style can also be defined in the dialog properties. See *Defining the Dialog Properties*.

To apply a style to a control or dialog

1. In the dialog, select the control(s) to which you want to apply a style.

   If you do not select a control, a dialog style can be applied.

2. Invoke the context menu and choose **Apply Style** > ***stylename***.

   You can also apply a style named "Default". In this case, the control or dialog is reset to the original settings as defined in the resource editor.

   Or:
   Choose the desired style from the font toolbar.

Saving the DDT File

DDT stands for Dialog Definition Table.

The DDT contains all information that has been defined using the SDK (such as font and color, list box properties, correspondence between character screen fields and dialog controls).

This file is always created in the same folder as the DLL containing the dialogs. It receives the same name as the DLL. If you copy the DLL to another location, you must also copy the DDT to this location.

To save the DDT file

- From the **File** menu, choose **Save DDT File**.

  Or:
  Choose **Save DDT** from the context menu that is invoked by pressing the right mouse button.

Building the BDD File

BDD stands for Binary Dialog Definition.

The DDT is the basis for building the BDD file. The BDD contains the DLL and DDT in binary format. When you define the extended detection rules for a character screen (see below), you have to specify the name of the BDD file.

To build the BDD file

- From the **Extended** menu, choose **Build BDD**.

  Or:
  Choose **Build BDD** from the context menu that is invoked by pressing the right mouse button.

  Information about the build process is now shown in the output window.

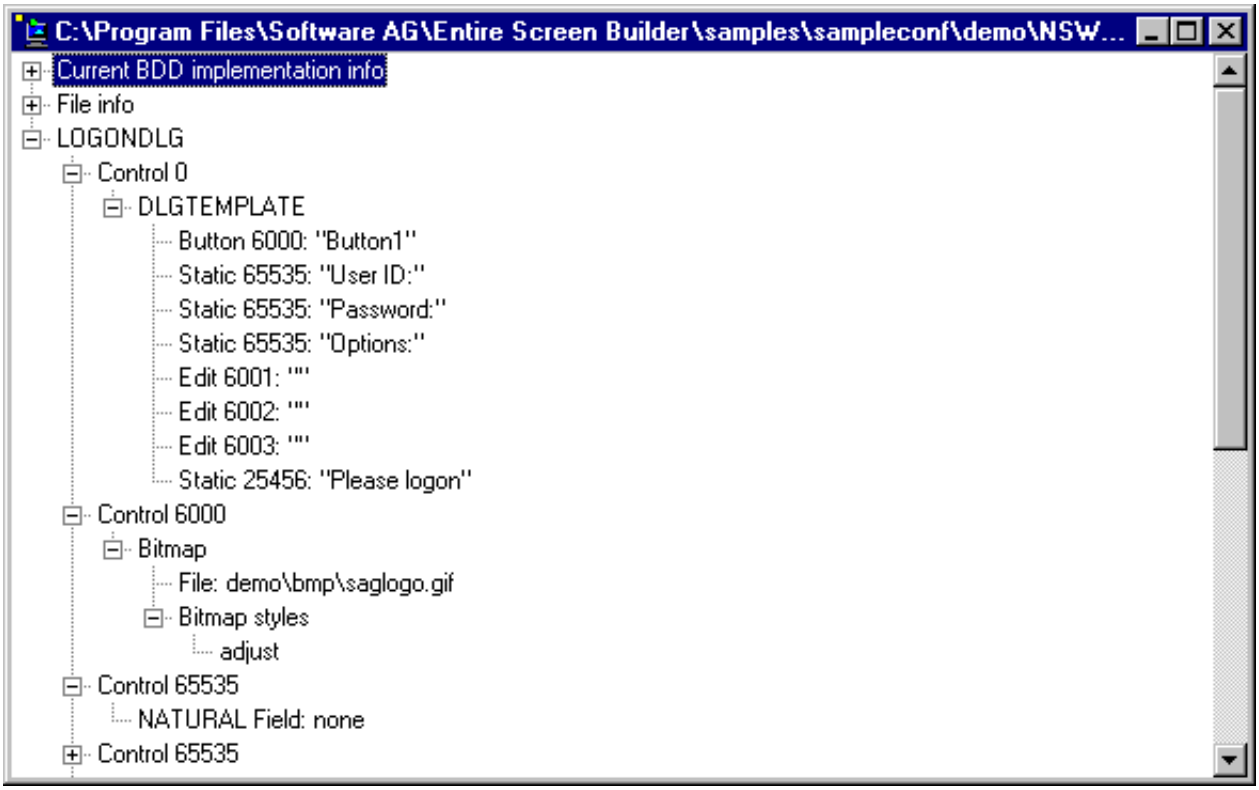
  When two or more IDs with the same numbers are detected, an error message such as the following is shown in the output window:

  ```
  Error: duplicate ID 512 found in dialog "MYDIALOG", controls tab order "10" and "20".
  ```

  The numbers given for the tab order refer to your specifications in the resource editor. This is the order in which the TAB key moves the input focus from one control to the next within a dialog box. Microsoft Visual Studio, for example, provides the menu command **Tab Order**. With this command, you can easily identify the controls that caused the error: just check the number in the upper-left corner of a control and compare it with the numbers in the error message.

  Note:
  If you use another resource editor, a **Tab Order** command may not be available or it may have another name.

Using *BDDView.exe* in Entire Screen Builder's *Bin* folder, you can view the contents of a BDD file. For example:



Detecting the Screen on which the Extended Rules are to be Applied (Single Screen)

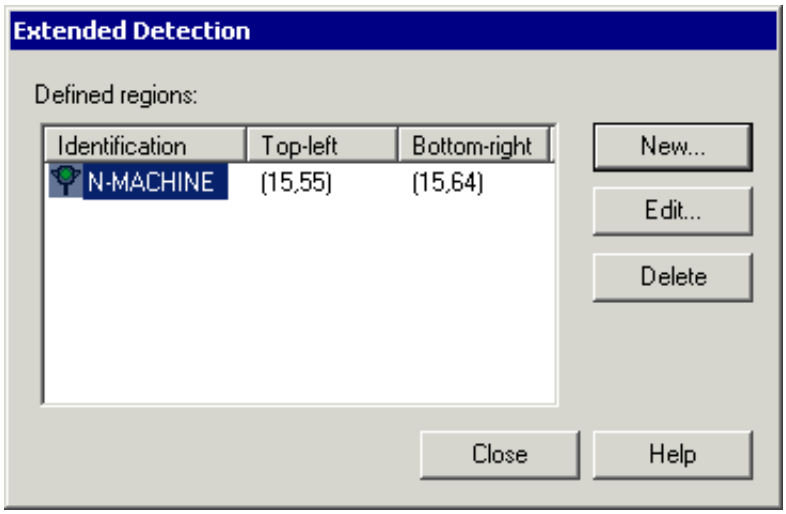
When the BDD has been built, you have to associate each extended dialog with the character screen for which it is to be displayed.

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.

To display all defined detection rules for a single screen

- From the **Extended** menu, choose **Detection** > **Single Screen**.

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



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

New	Add a new detection rule (see below).
Edit	Modify the selected detection rule. Alternative: double-click a detection rule.
Delete	Delete the selected detection rule. Alternative: choose **Delete** from the **Edit** menu or press DEL to delete the selected detection rule.

To associate a dialog with a character screen

1. Make sure that the resource file that is to be used as a template is shown in the scope window.
2. Display the Extended 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.

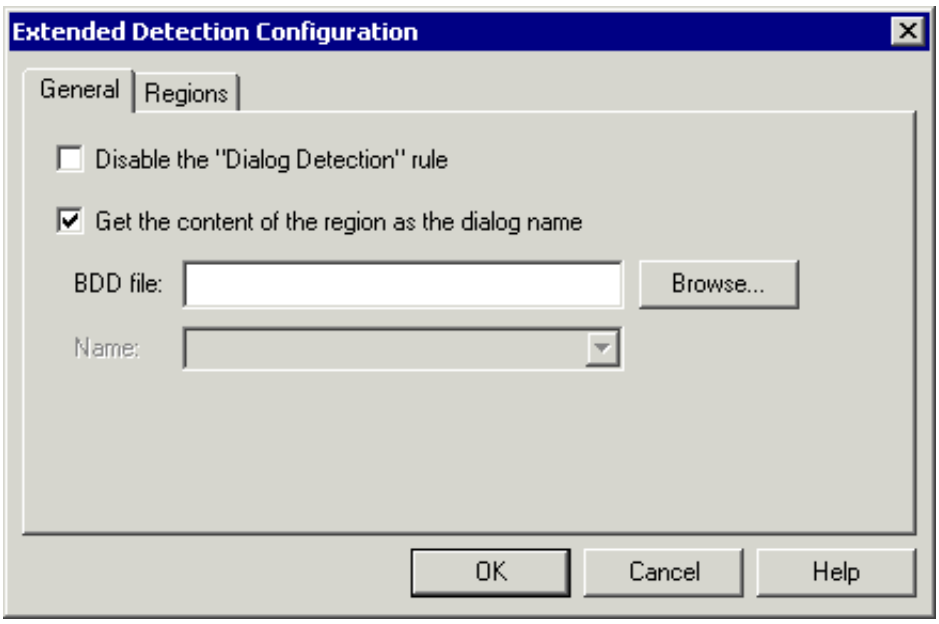
   The Extended Detection Configuration dialog box appears.
5. Specify all required information as described below.
6. Choose the **OK** button.

Overview of Extended Detection Configuration Options

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

- General
- Region

General



Disable the Dialog Detection rule

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

Get the content of the region as the dialog name

When this check box is selected, the server will use the content of the first region that is defined in the BDD file as the dialog name (see the description of the Regions page).

Important:
When subregions have been defined, the content of the region is only used as the dialog name if all defined subregions are also found. The subregions itself are not used as dialog names.

This feature is helpful, if the character screens always display their different names in the same region. In this case, you only have to define one rule.

BDD file

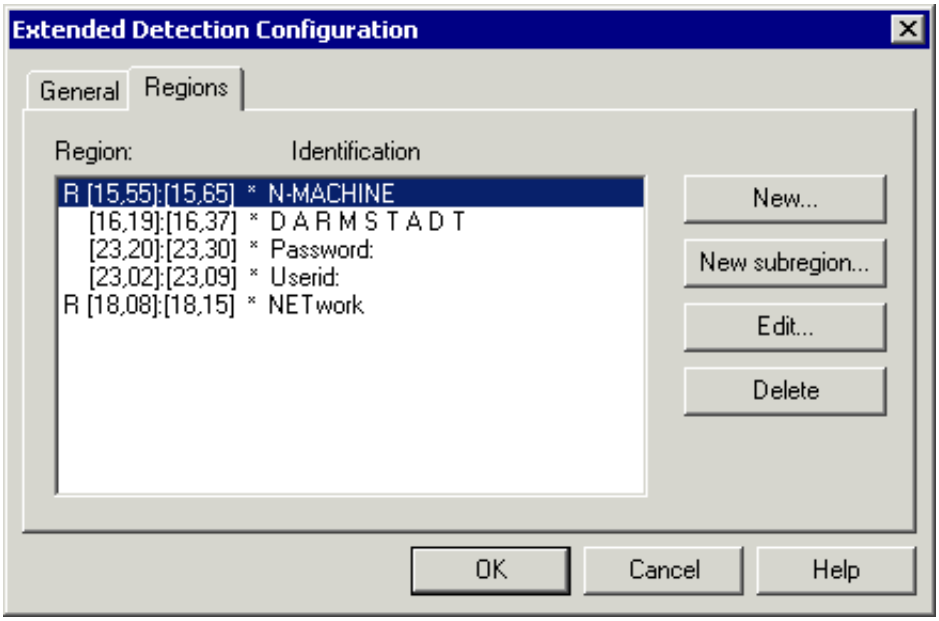
Specify the path to the BDD file containing the extended dialog for the character screen. You can also choose the **Browse** button to select the BDD file from the Open dialog box.

Name

When a BDD file has been specified, you can select one of the extended dialogs contained in the BDD file from this drop-down list box. This is the dialog that is to be shown instead of the character screen. This text box is dimmed if the **Get the content of the region as the dialog name** check box is selected.

Region

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.

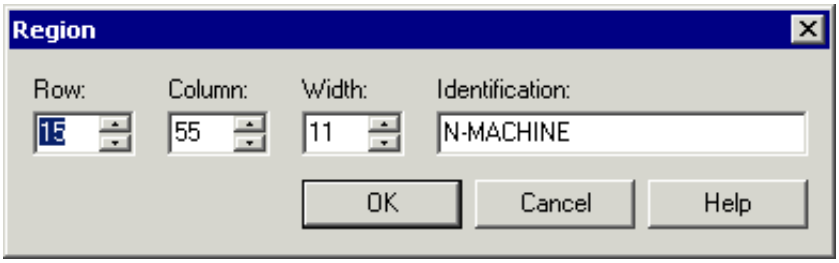
Example: with the above definitions, the screen is detected when region 1 *and* all defined subregions are found, *or* when region 2 is found.

The following command buttons are provided:

New	Define a new region for the current extended dialog.
New subregion	Define a new subregion for the selected region. For one region several subregions can be defined.
Edit	Modify the selected region or subregion. Alternative: double-click a region or subregion to modify it. See *Modifying a Region or Subregion* below.
Delete	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 the extended dialog are to be applied.

Using the Multi Screen Feature

The multi screen feature enables you to combine several character screens into a single dialog. Intermediate screens can be hidden so that they cannot be seen by the user. Data can be collected from several screens and can then be presented in a different way.

This chapter covers the following topics:

- Multi Screen Scenarios
- Required Steps
- Displaying All Defined Multi Screen Rules
- Detecting the First Character Screen
- Defining the BDD File and the Dialog
- Defining the Green Screens
- Defining the Regions and Subregions for Each Screen
- Defining the Options for Each Screen
- Error Handling

Multi Screen Scenarios

The multi screen feature uses a dialog for which extended rules have been defined. You can use this feature for several purposes:

- Navigation
- Output Collection from Several Screens (Many to One)
- Input to Many Screens (One to Many)

Navigation

You can navigate from a start screen to a final screen without showing the middle screens.

In the rule, you define the following for each screen: the data that have to be entered in the fields of the screen and the function key that is to be used to proceed to the next screen. The data for the first screen can be variable data, for example, the user ID and password for logon. They can be entered by the user in the dialog.

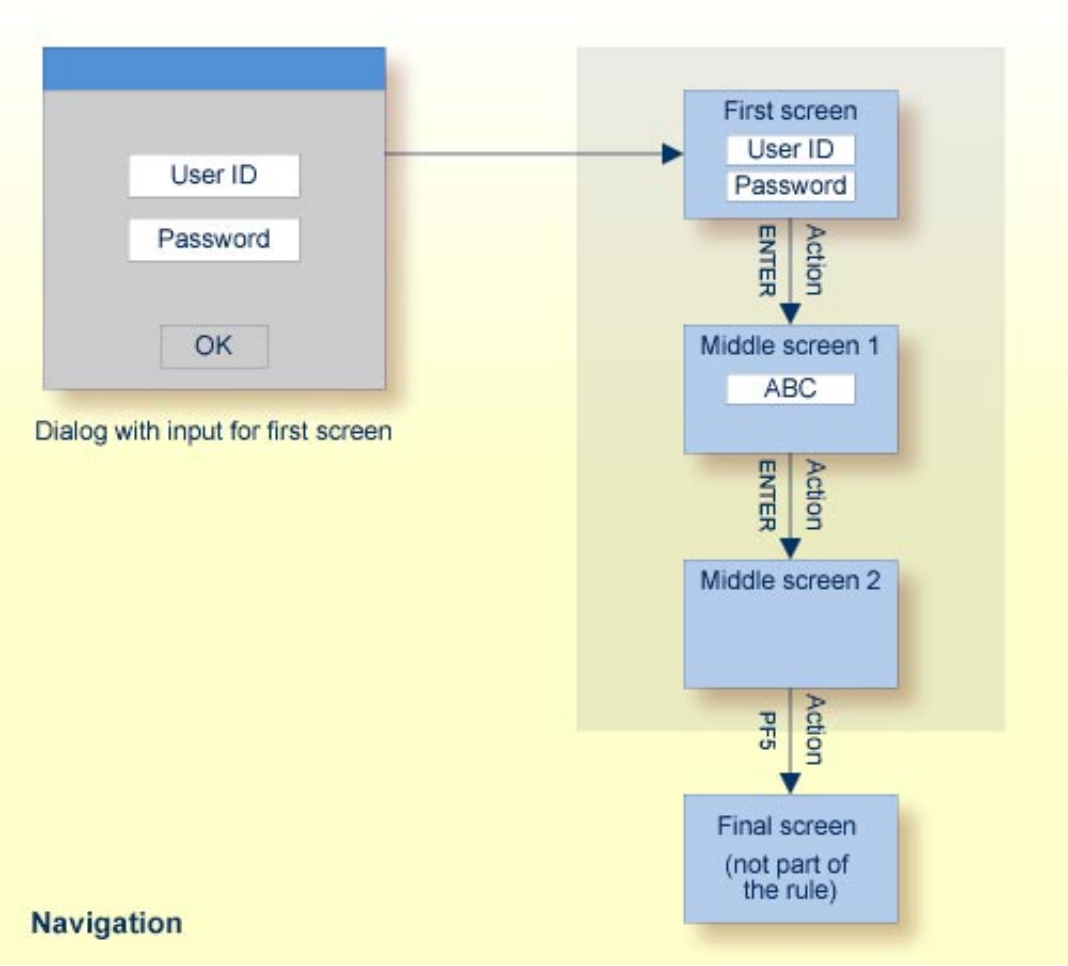
The dialog is shown when the first character screen is detected. This dialog either contains input fields or no input field at all. The input data from the dialog are entered into the character screen.

The input data and function keys for the middle screens are automatically entered as defined in the rule.

The final screen is displayed in the usual way. It is not part of the multi screen rule.

Example

An example for this navigation is the logon to a specific application where the same steps are always required. There is a first screen in which you have to enter your user ID and password. On the next screen, you always have to enter the same application ID and press ENTER to proceed to the application's welcome screen. On this screen, you press PF5 to start your actual work.



Output Collection from Several Screens (Many to One)

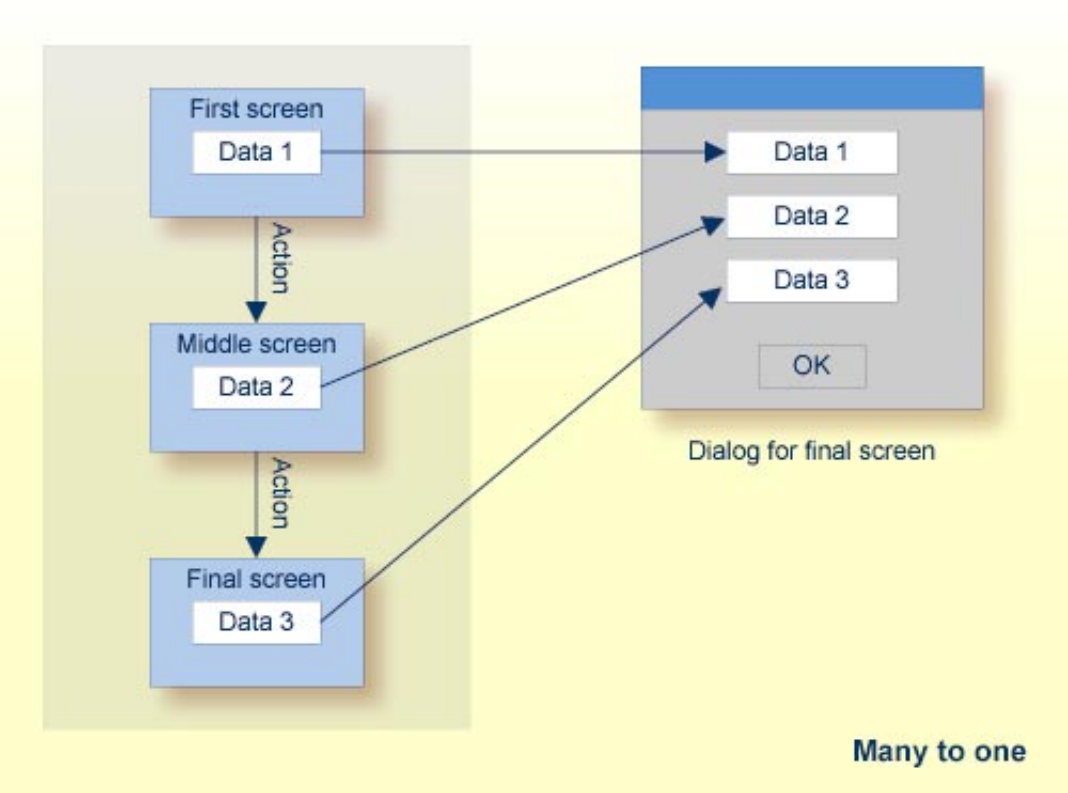
You can collect data from several character screens and show this data in a single dialog. We call this feature "many to one".

From a start screen, you want to navigate to a final screen without showing the middle screens. This is navigation as described above. In addition, data from the first screen and all middle screens are collected and merged into the final screen. The middle screens are not shown. The dialog is only shown for the final screen. The data collected from the previous screens are filled in.

The dialog for the final screen may also contain, for example, edit boxes and push buttons. However, these controls can only be linked with the final screen. The data collected from the previous screens are for display only.

Example

The "many to one" feature is useful when the information you want to obtain from the application is spread across several screens which also contain information that is not needed at this time.



Input to Many Screens (One to Many)

You can create a single dialog which contains data for several character screens. We call this feature "one to many".

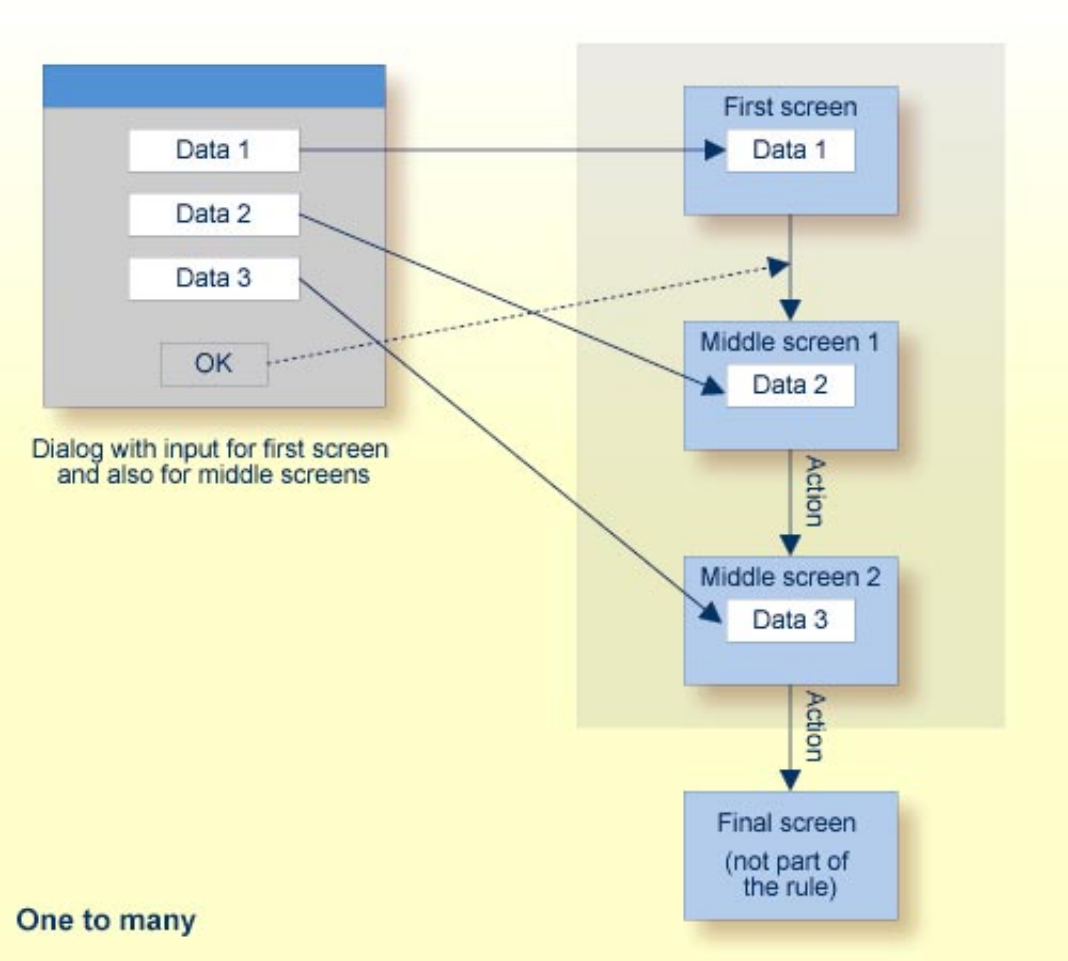
From a start screen, you want to navigate to a final screen without showing the middle screens. Variable input (supplied by the user) is required on several screens.

A dialog is shown for the first screen. It contains input fields for the data that is to be used on the first and subsequent screens. The dialog contains only one button with the host function that is used to proceed from the first screen to the next screen. Subsequent screen navigation is controlled by the actions that have been defined for each screen in the multi screen rule.

The final screen is displayed in the usual way. It is not part of the multi screen rule.

Example

Many applications provide selection criteria on more than one screen for the retrieval of information requested by the user. The "one to many" feature enables the user to enter all required selection criteria on a single screen so that the required information is obtained in one step.



Required Steps

1. Identify the character screens (also called "green screens" in this documentation) for which you plan to write a multi screen rule. Note down all required input data for a screen and the action (function key) that is used to proceed to the next screen.

2. Capture and save the screen files for all character screens that are to be detected (see *Using Screen Files*).

3. Use a resource editor to create a dialog for the required multi screen scenario.

4. Use the SDK to open the DLL containing the dialog that you have previously created with the resource editor.

5. Open the dialog that is to be used instead of the green screens. Each dialog can only be used for one multi screen rule.

6. Enable the multi screen option in the dialog properties (see *Defining the Dialog Properties*).

7. For each control in the dialog, specify the required control properties (see *Defining the Control Properties*):

   - On the Field Identification page, define the corresponding field on the character screen.

   - On the Screen Number page specify the number of the character screen on which the corresponding field is located.

8. Save the DDT file and build the BDD file (see *Saving the DDT File* and *Building the BDD File*).

9. From the **Extended** menu, choose **Detection > Multi Screen** to define the multi screen rule and the detection rules for all character screens that are to be used. This is explained in the remainder of this document.

10. Build the BDD file once more.

11. Decide whether the session is to be disconnected automatically when the multi screen feature fails in the viewer. See the description of the Server page of the Client Control Properties dialog box.

Note:
See also: *Rules Analysis* which is a useful feature for developing multi screen rules.

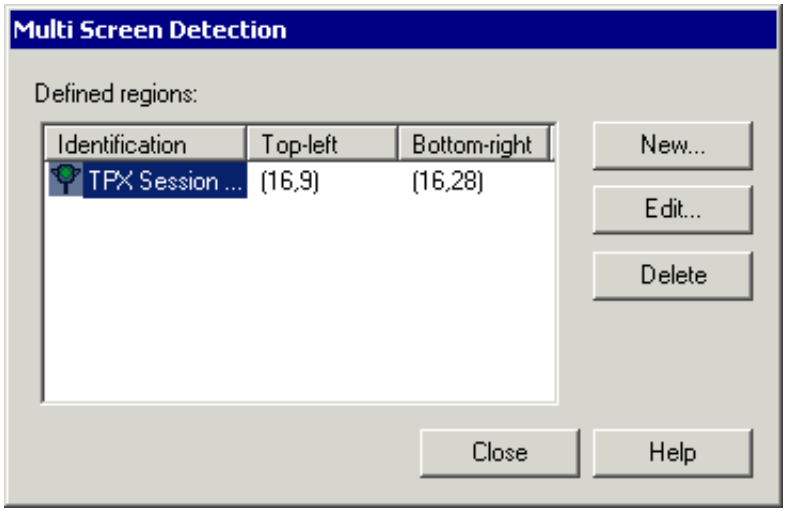
Displaying All Defined Multi Screen Rules

Each rule that is defined in the Multi Screen Detection dialog box (see below) applies to one set of character screens.

To display all defined multi screen rules

- From the **Extended** menu, choose **Detection** > **Multi Screen**.

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



When multi screen rules have already been defined, they are shown in this dialog box. When a rule is selected, the scope window shows the screen file that was open when the detection rule has been defined. The screen file contains an outline for the defined region that is used to detect this rule.

The identification that is shown for a multi screen rule applies to the first screen that is to be detected.

The color red in the symbol to the left of a rule 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:

New	Add a new multi screen rule and define the first screen to be detected. See *Detecting the First Character Screen*.
Edit	Modify the selected rule. Alternative: double-click a rule.
Delete	Delete the selected rule. Alternative: choose **Delete** from the **Edit** menu or press DEL to delete the selected rule.

Detecting the First Character Screen

When you add a multi screen rule, you first have to define the detection rule for the first screen.

Note:
The detection rules for all other screens are defined later. See *Defining the Green Screens*.

To define the detection rule for the first character screen

1. Make sure that the screen file that is to be used as a template is shown in the scope window.

   The name of the screen file that is used to define the detection rule for a character screen is stored in the rule. When the rule is later selected in the Multi Screen Detection dialog box, the stored screen file name is used to open the corresponding screen file automatically.

2. Display the Multi Screen Detection dialog box as described above (see *Displaying All Defined Multi Screen Rules*).

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 first screen.

   Note:
   It is also possible to define further regions and subregions for detection. See *Defining the Regions and Subregions for Each Screen*.

   The Multi Screen Detection Configuration dialog box appears.

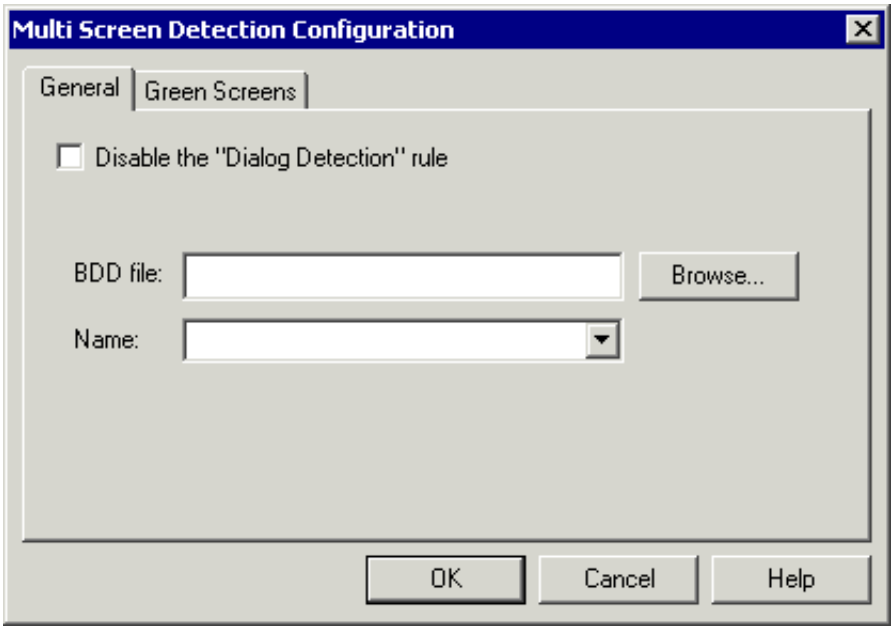
5. Specify all required information as described in the following sections.

   - *Defining the BDD File and the Dialog*
   - *Defining the Green Screens*

6. Choose the **OK** button.

Defining the BDD File and the Dialog

The dialog box containing the page below appears when you choose the **New** or **Edit** button in the Multi Screen Detection dialog box.



Disable the Dialog Detection rule

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

BDD file

Specify the path to the BDD file containing the dialog that has been created for your multi screen scenario. You can also choose the **Browse** button to select the BDD file from the Open dialog box.

Name

When a BDD file has been specified, you can select one of the dialogs contained in the BDD file from this drop-down list box. This is the dialog that is to be shown instead of the green screens.

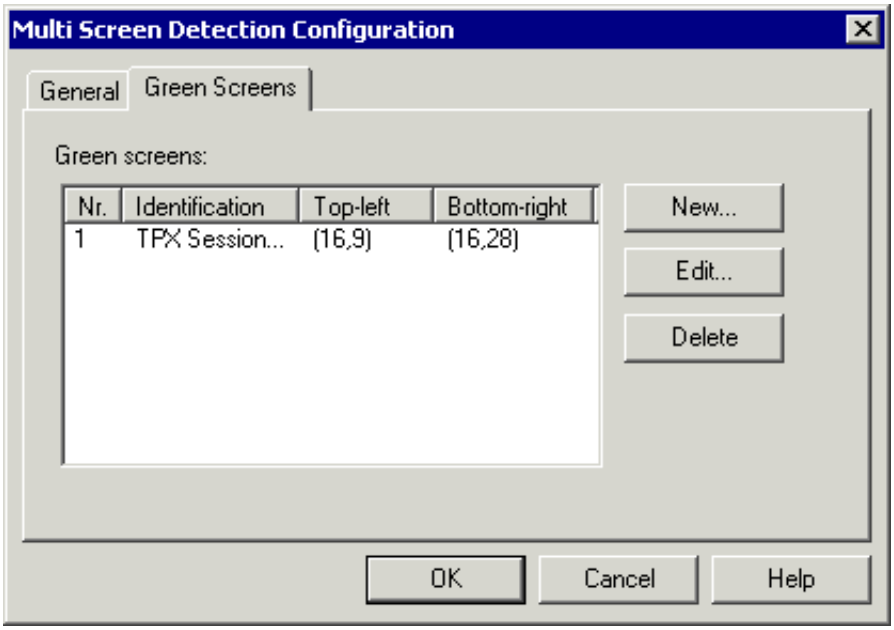
Important:
One dialog can only be used for one multi screen rule.

Defining the Green Screens

The dialog box containing the page below appears when you choose the **New** or **Edit** button in the Multi Screen Detection dialog box.

The detection rule for the first screen is the detection rule for the multi screen rule itself (indicated by the number 1). To complete the definition for the first screen, select screen number 1 and choose the **Edit** button. Specify all required information on the General page of the Green Screen dialog box. See *Defining the Options for Each Screen* for further information.

After this, you have to define the detection rules for all other green screens that are to be used. Define the screens in the same sequence in which they are invoked on the host.



When you select a green screen in the above dialog box, the scope window automatically shows the screen file in which the detection rules for this screen have been defined.

The following command buttons are available:

New	Add a new green screen and define its detection rules. See below.
Edit	Modify the selected green screen. Alternative: double-click a green screen.
Delete	Delete the selected green screen. Alternative: choose **Delete** from the **Edit** menu or press DEL to delete the selected green screen.

To add a green screen and define the detection rule

1. In the Multi Screen Detection Configuration dialog box (see above), choose the **New** button to add a new green screen.

   The Green Screen dialog box appears. The Region page is shown first.

2. Make sure that the screen file that is to be used as a template is shown in the scope window. To open another screen file, choose the **Open screen file** button in the Region page. You can then select the required screen file from a dialog box.

3. In the Region page, choose the **New** button.

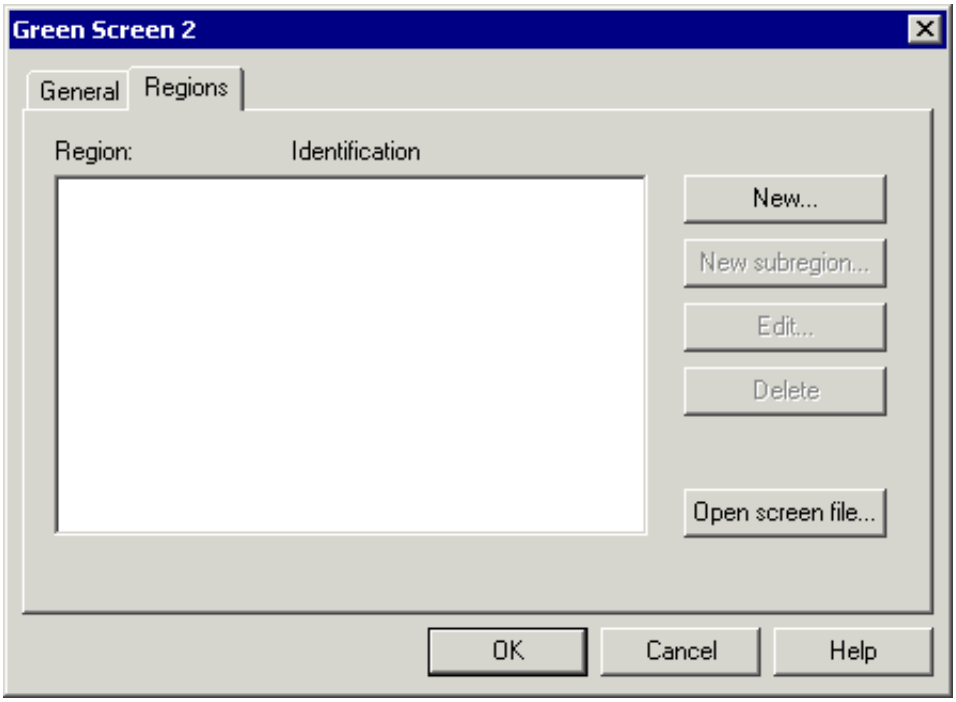
4. In the scope window, use the mouse to select the region which contains the string that is to be used to detect the screen. See *Defining the Regions and Subregions for Each Screen* for further information.

5. Invoke the General page of the Green Screen dialog box and specify all required information. See *Defining the Options for Each Screen* for further information.

6. Choose the **OK** button.

Defining the Regions and Subregions for Each Screen

The dialog box containing the page below is shown when you add a new green screen or when you edit an existing green screen in the Multi Screen Detection Configuration dialog box.



The Regions page is used to define regions and subregions, i.e. unique strings which appear on each green screen that is to be detected. You can define, for example, that the screen is detected when region 1 *and* all defined subregions are found, *or* when region 2 is found.

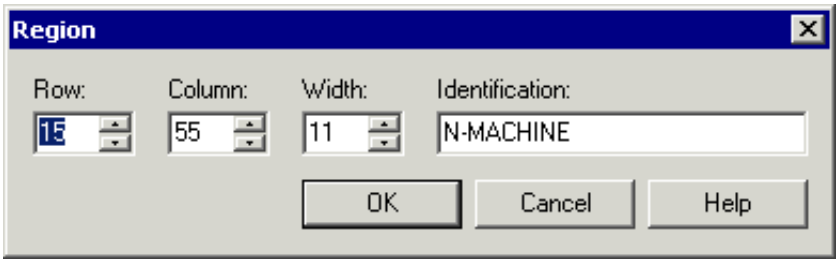
On this page, a defined region is indicated by the letter "R" at the beginning of a line. Lines which do not begin with the letter "R" apply to subregions.

The following command buttons are provided:

New	Define a new region for the current screen. You have to select the required region in the screen file which is shown in the scope window.
New subregion	Define a new subregion for the selected region. You have to select the required subregion in the screen file which is shown in the scope window. For one region several subregions can be defined.
Edit	Modify the selected region or subregion. Alternative: double-click a region or subregion to modify it. See *Modifying a Region or Subregion* below.
Delete	Delete the selected region or subregion. When you delete a region, all of its subregions are automatically deleted.
Open screen file	Select a new screen file to be displayed in the scope window.

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

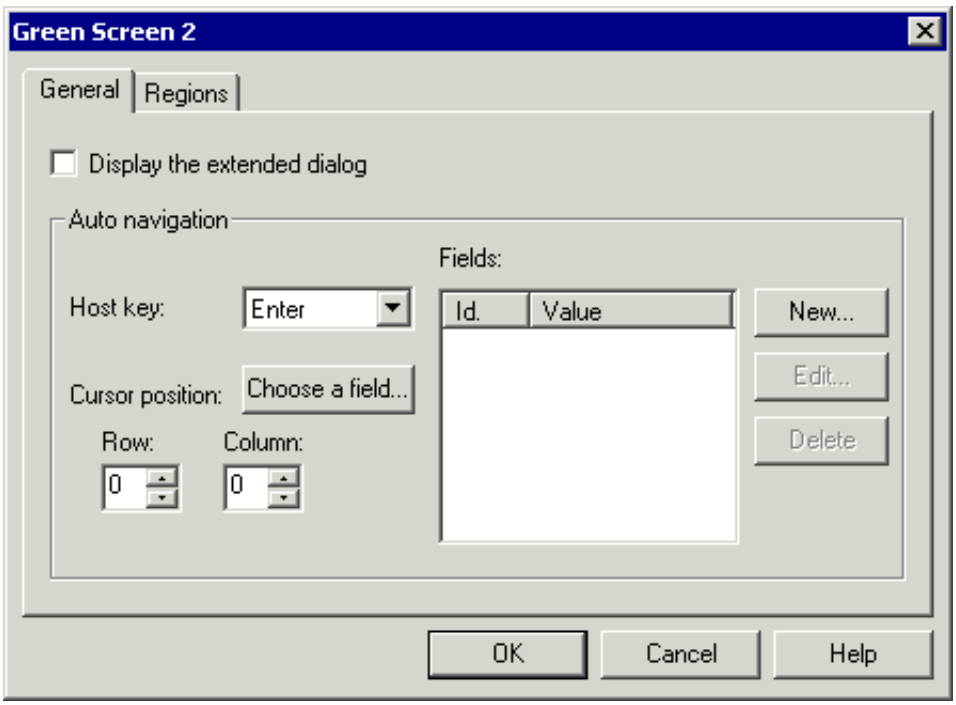
The position of the string that is to be used to detect the screen. This is the region that has been selected using the mouse. Using the spin buttons, you can manually adjust the values.

Identification

This is the string that has been selected using the mouse. This string is used to detect the screen.

Defining the Options for Each Screen

The dialog box containing the page below is shown when you add a new green screen (after a region has been defined) or when you edit an existing green screen in the Multi Screen Detection Configuration dialog box.



Display the extended dialog

When this check box is selected, the dialog will be shown in the viewer when this green screen is detected. In this case, the options for auto navigation are not available.

When this check box is not selected, the dialog will not be shown in the viewer. In this case, you have to specify the required auto navigation options. When the current green screen is detected, these options are used to navigate automatically to the next green screen.

Host key

Select the host key that is to be sent to the host in order to navigate to the next screen.

Cursor position / Row / Column

Specify the row and column into which the cursor must be placed in order to navigate to the next screen. You can either define an input field or an output field.

When a screen file is shown in the scope window, you can also choose the **Choose a field** button to select the required field.

Fields

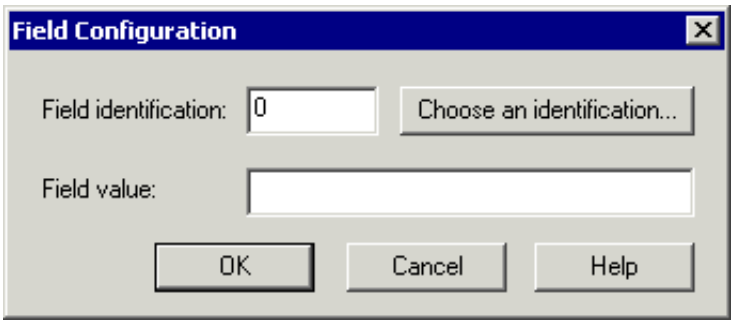
To navigate to the next host screen, it may be required to move data to specific input fields.

New	Define the value for a specific field. See *Sending Data to an Input Field on the Host* below.
Edit	Modify the selected field. Alternative: double-click a field.
Delete	Delete the selected field. Alternative: choose **Delete** from the **Edit** menu or press DEL to delete the selected field.

Sending Data to an Input Field on the Host

This dialog box appears when you choose the **New** or **Edit** button on the General page of the Green Screen dialog box (next to the list box containing the defined fields).

It is used to move data to a specific input field.



Field identification

Either type the field ID in the text box or choose the **Choose an identification** button to select the required field in the scope window.

Field value

Specify the value that is to be moved to the input field.

Error Handling

There is no specific error handling available for the multi screen rule in this version of Entire Screen Builder.

When the first character screen of a multi screen rule is not detected, the rule will not be applied.

If the next middle screen in a sequence is not detected, the multi screen rule will be terminated and the screen will be displayed with normal screen processing. This means that the rule is also terminated if the same middle screen appears a second time.