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.

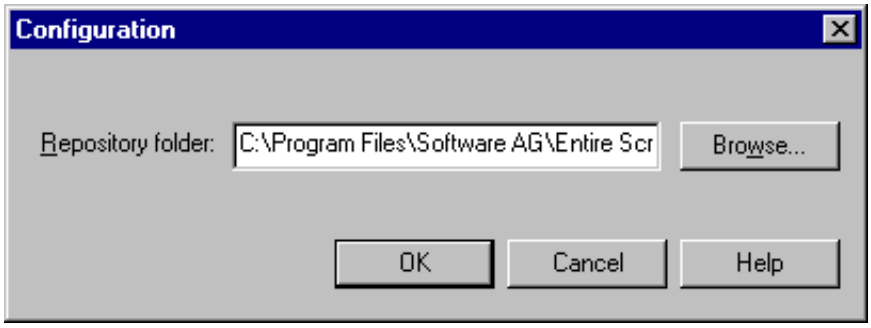
You can also define the location of the rules repository with the System Management Hub. See *GUI Version* in Entire Screen Builder's *System Management Hub* documentation.

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.

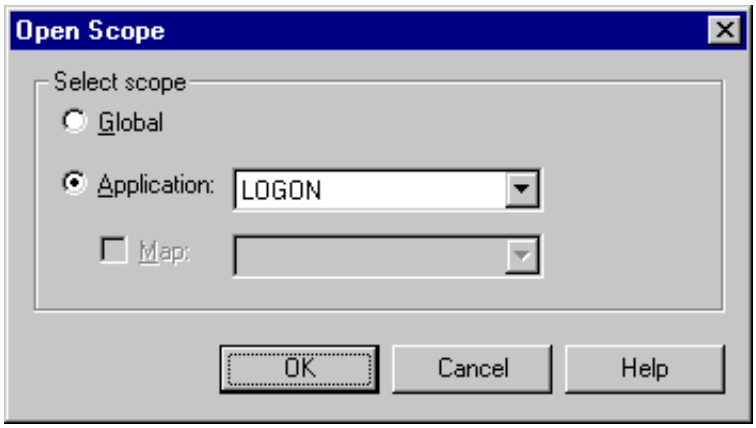
Important:
Before you start defining rules, make sure that the required scope has been opened.

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**.

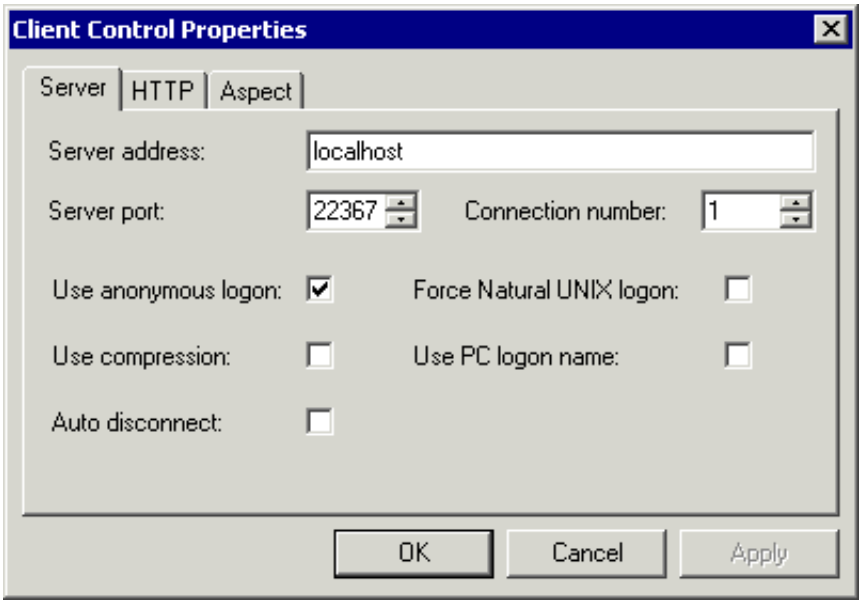
   The Client Control Properties dialog box appears. You can define properties on the following pages: Server, HTTP and Aspect. Information on the properties is provided below.

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.

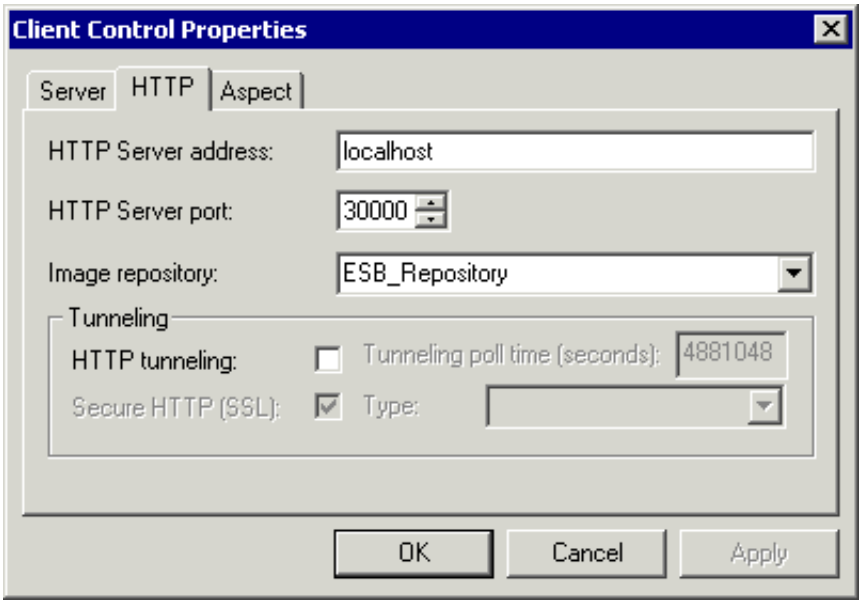
Property names in above dialog box	**Property names for user exits**
Server address	`Server`
Server port	`Port`
Connection number	`Connection`
Use anonymous logon	`AnonymousLogon`
Force Natural UNIX logon	`UnixLogon`
Use compression	`Compressed`
Use PC logon name	`UsePCLogonName`
Auto disconnect	`AutoDisconnect`

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.



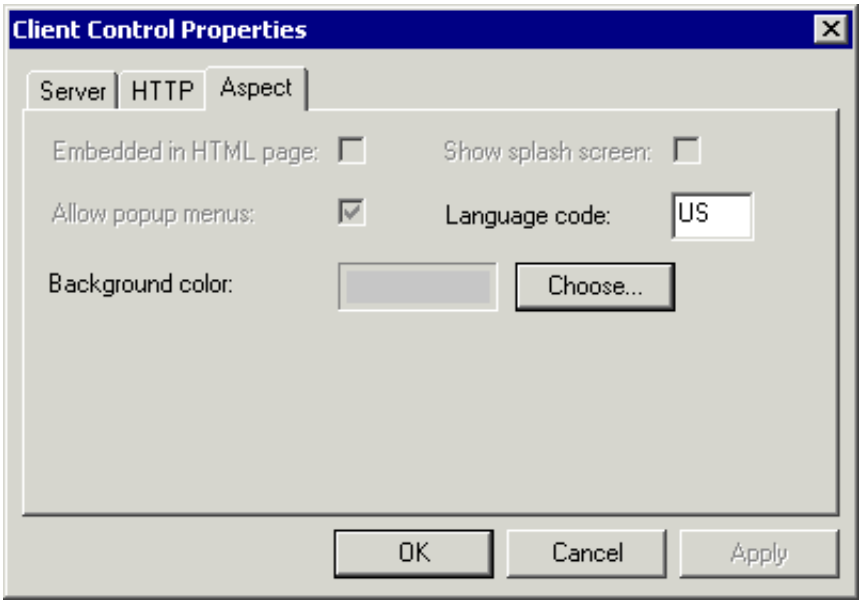
Note:
The drop-down list box **Image repository** contains all aliases defined in the System Management Hub. See *Alias List* in Entire Screen Builder's *System Management Hub* documentation.

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.

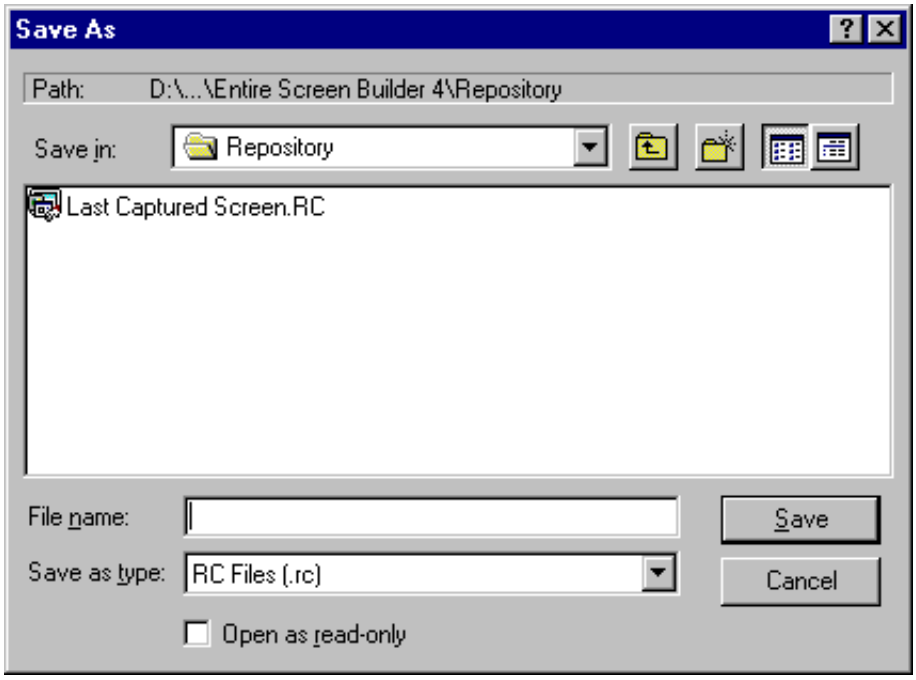
Saving a Host Screen as a Resource File

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.

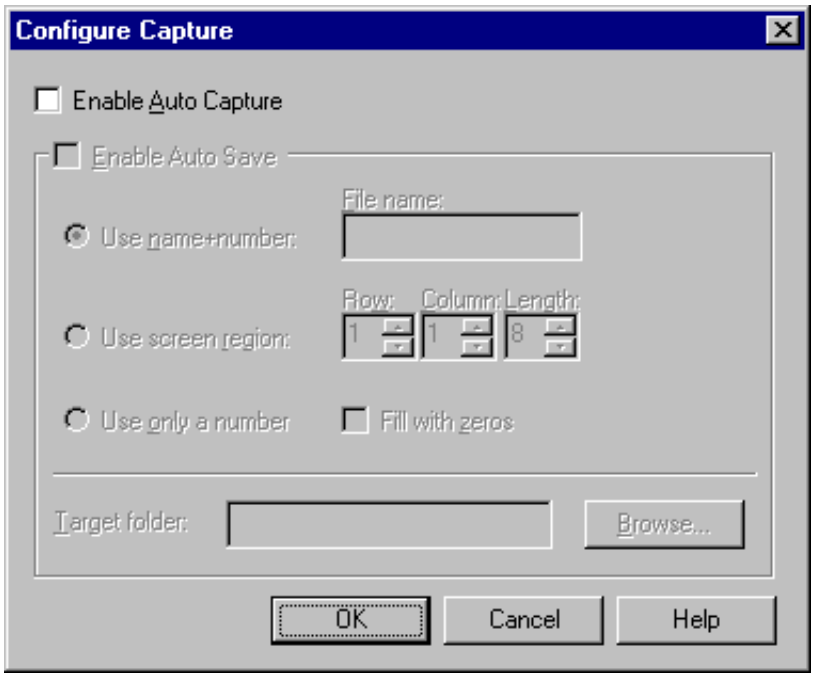
Capturing and Saving a Screen File Automatically

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.

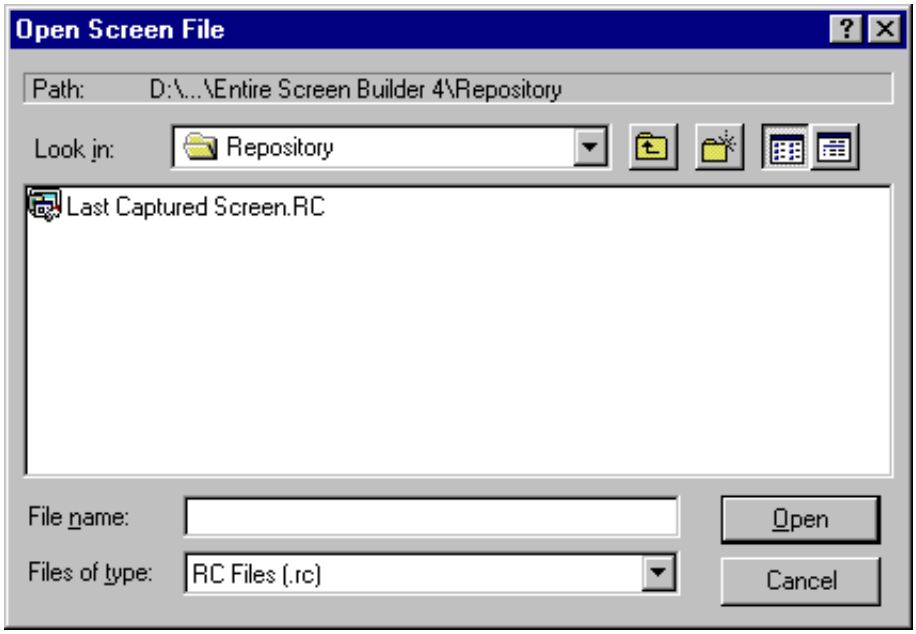
Important:
You can only open a screen file when a scope has been opened.

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.

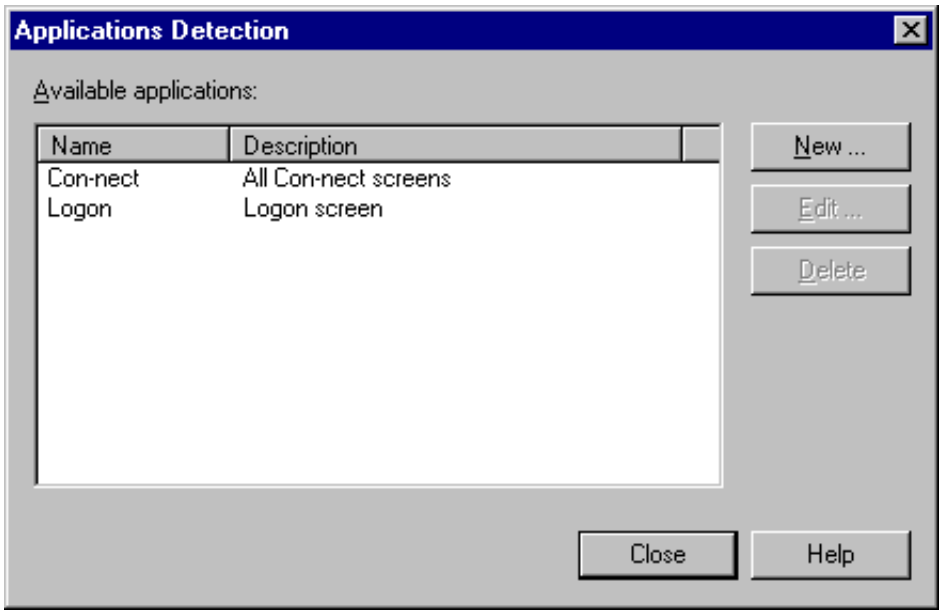
To display all defined applications

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.



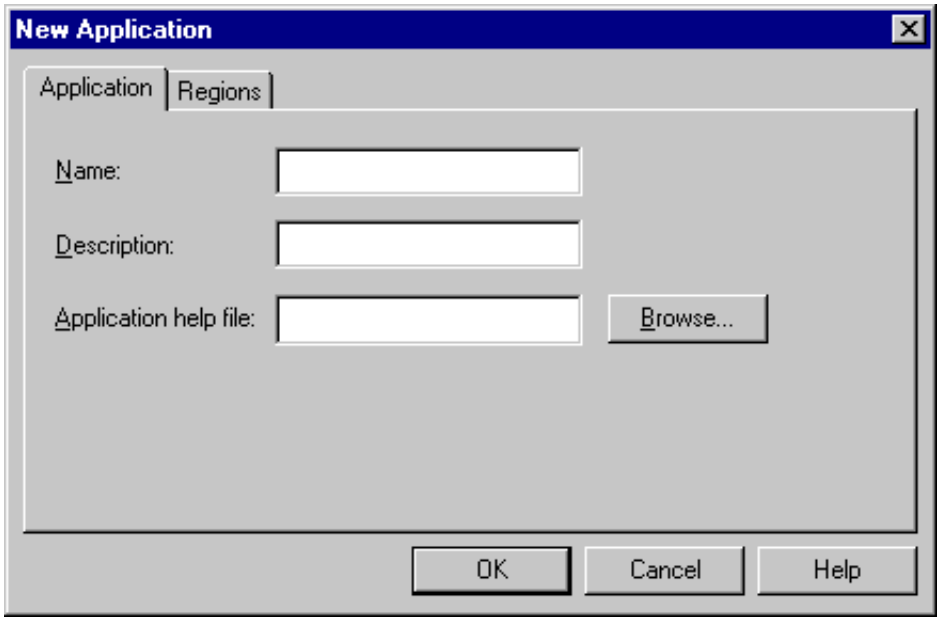
The following command buttons are provided:

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.

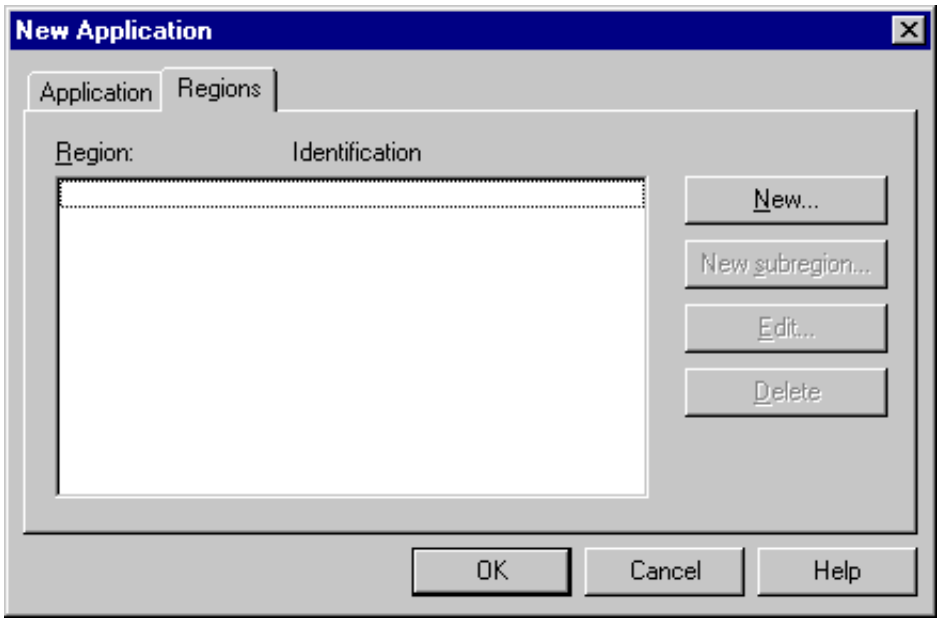
   A folder with this name will be created. All rules that you define for this application will be written to the *Defaults.ini* file in this folder. It is not possible to modify the application name once the application has been added.

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.

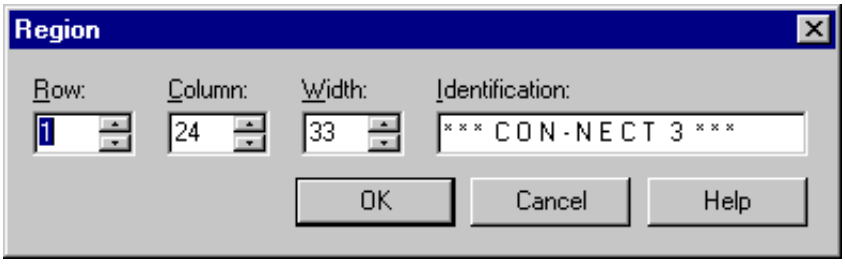
11. Choose the **OK** button to save your modifications.

    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

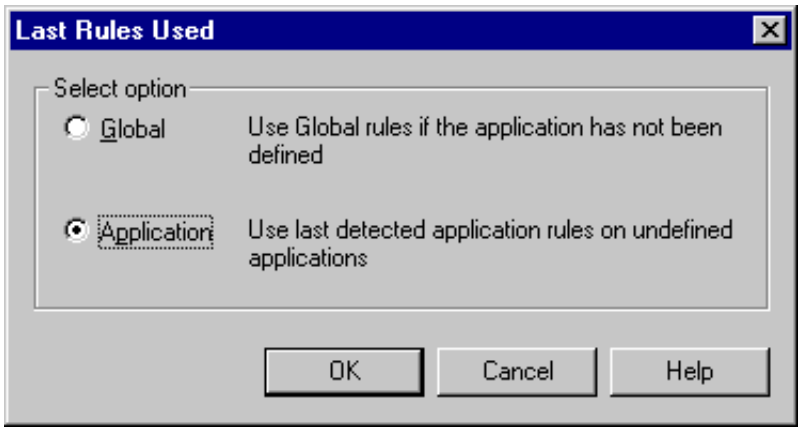
You can define the rules that are to be applied to a character screen for which an application has not been defined. By default, the rules of the last detected application are applied.

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.