# General Information

This chapter applies to the Entire Screen Builder user exits. It provides general information on the following:

- Server Side
- Viewer Side

---

## Server Side

An Entire Screen Builder user exit is a Dynamic Link Library (DLL) with a set of user-written functions that comply with the interface definition described below. You can specify a user exit DLL for global scope, every application scope and every map scope.

When a User Exit rule has been defined for a given scope, the viewer dynamically loads the DLL when this scope is entered and tries to find the user-exit functions `NSWBefore` and `NSWAfter` in the DLL.

For each new screen in this scope, the viewer calls the user-exit DLL two times: the first time before the screen is shown, and the second time after the user has entered data into the screen - just before the data are sent back to the host. The first call allows to influence the way a screen is displayed by the viewer. The second call gives the chance to process and modify the data entered by the user before they are sent to the host.

The DLL is unloaded when a given scope is left.

To create the DLL, use a development tool such as Microsoft Visual Studio.

The user-exit functions should be written in the C or C++ programming languages. They have to be compiled and linked to a DLL. The following user-exit functions should be implemented and exported in the DLL:

| **Function** | **Usage** |
|---|---|
| NSWBefore | Called each time when a screen is received from the legacy application. This allows to process the screen received before it is processed and displayed by the viewer. |
| NSWAfter | Called each time the viewer is about to send data to the legacy application. This allows to process the data typed by the end-user before the viewer sends them to the legacy application. |

If one of the above functions is not exported in the DLL, the viewer will detect this and will never try to call it. However, there will be no error message telling about this fact.

See *Server Functions* for further information.

# Viewer Side

Entire Screen Builder’s Web Viewer and Terminal Viewer are Microsoft ActiveX controls. Thus, COM interfaces can be used to

- access the defined methods and properties, and to
- receive events.

**Note:**
A COM interface is not available for the Windows Viewer.

The COM interface for the Web Viewer is different from that used for the Terminal Viewer.

See the sections *Web Viewer Client Functions (API)* and *Terminal Viewer Cient Functions (API)* for further information.