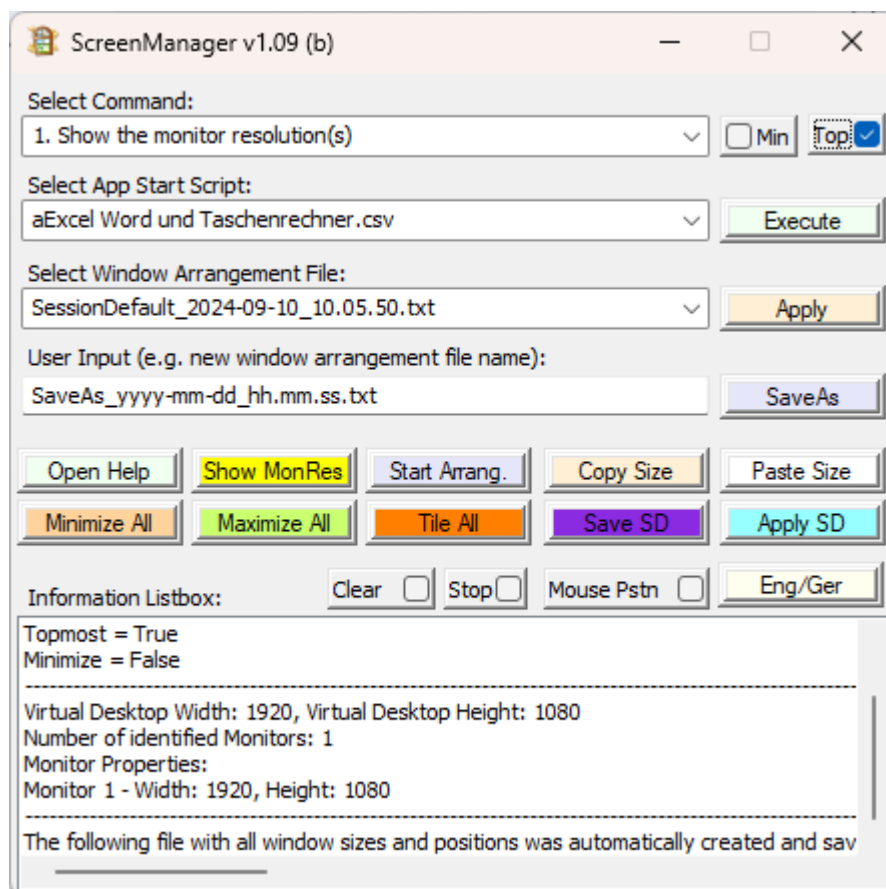


## Screen Manager Operating Instructions

1.	Summary of the most important Features .....	2
2.	Description of the Operating Elements on the Program Interface .....	4
3.	Description of the built-in Commands .....	5
4.	Description of the User Settings .....	6
5.	Revision List.....	6



## 1. Summary of the most important Features

The ScreenManager SCM is a program written in C++ that is able to save and restore Windows window arrangements. When saving, the current window arrangement is saved in a text file. There are several commands to restore a saved window arrangement from a file. The following properties are taken into account:

- Window size x/y (measured in pixels)
- Window position x/y (top left corner of the window)
- Window position z (stacked windows)

### Note on the z-Position:

Open programs are usually displayed via a visible window. For example, you can see all five overlapping program windows if the largest window is at the bottom (e.g., z=1) and the subsequent windows follow in descending size (e.g., z=2 to z=5). If you now click on the largest window at the bottom, it moves to the foreground (e.g., z=5) and covers the other four windows, which you can then no longer see. In real Windows life, the numbers for z can sometimes exceed 100 or even several hundred, even though there are fewer than ten visible windows.

I originally programmed this application for my former company to test software. The test engineers had at least three, some even five screens. Aligning the windows until they were ready for testing could easily take two minutes. Once the desktop or e-mail program was clicked on, the painstakingly arranged layout was gone again and the clicking started all over again. For the purpose described, the SCM was almost irreplaceable. The SCM is less valuable for normal users, but it can still be a helpful tool.

### Here is a Summary of some useful Functions:

- Save window arrangements at the touch of a button. You can give each window arrangement its own name, but you do not have to (automatic generation of file names without overwriting other, already saved arrangements).
- Restore a window arrangement at the touch of a button.
- Creation and storage (Excel→ CSV) of start scripts (applications, files plus window arrangement) and their call during program execution.
- Display of the mouse position in the SCM list box by ticking a checkbox.
- Display of all open windows (up to approx. 90) as tiles for an overview.
- Assignment of the TOPMOST status (permanently at the top in the z-direction) to a previously selected window.
- Copying and pasting window sizes.
- Start all subdirectories and files via the program interface.

At this point I would like to confess what the SCM cannot do, namely start files in a window arrangement after a reboot. This would require the Windows API function NtQuerySystemInformation, which does exist:

## Screen Manager








```
__kernel_entry NTSTATUS NtQuerySystemInformation(  
    [in] SYSTEM_INFORMATION_CLASS SystemInformationClass,  
    [in, out] PVOID SystemInformation,  
    [in] ULONG SystemInformationLength,  
    [out, optional] PULONG ReturnLength  
);
```

On the Microsoft Learn website (formerly MSDN), 16 of the current 83 system information classes are described. This means that 66 of these classes are undocumented. The information about open files could be taken from the SystemHandleInformation class, which, however, belongs to undocumented information classes and for which access is blocked by Windows. The way to unlock this, which some programmers have taken, is to program a pseudo hardware driver. Skipping this hurdle was too time-consuming for me. However, the start scripts are hopefully a passable alternative solution.

### Further Notes:

1. After a PC restart, all application windows contained in an arrangement file have to be started manually (refer to the confession above).
2. After a PC restart, it is also possible that one or more new windows are present which are not contained in the saved arrangement file. All these windows are recognized and minimized.

The Screen Manager is portable, i.e., it makes and requires no entries in the Windows registry. Its directory structure is as follows:

 _AppStartScripts	06.09.2024 09:32
 _Arrangements	06.09.2024 09:32
 _Exports	06.09.2024 09:45
 _Help	06.09.2024 09:30
 _License	06.09.2024 09:30
 _Settings	06.09.2024 09:30
 ScreenManager v1.09 (a).exe	06.09.2024 09:43

The \_AppStartScripts and \_Arrangements subdirectories contain exactly what is written on them. The \_Exports folder contains the text files generated and exported by some commands. The help folder contains the German and English help. The SCM is freeware. Therefore, the \_License subdirectory with a sample file only exists out of habit. The settings file is stored in the \_Settings folder.

## Screen Manager

### 2. Description of the Operating Elements on the Program Interface

Control element	Description
<b><u>Combobox</u>: Select Command</b>	Selection of one of 28 commands. Direct execution without a button.
<b><u>Combobox</u>: Select App Start Script</b>	Select app start script.
<b><u>Combobox</u>: Select Window Arrangement File</b>	Select arrangement file.
<b><u>Input field</u>: User Input</b>	User input, typically a file name.
<b><u>Listbox</u>: Information output</b>	Output of all types of information.
<b>Checkboxes:</b>	
<b>Top</b>	Set the SCM window to TOPMOST in the z arrangement.
<b>Min</b>	Minimizes the SCM window after the Apply button if the Min checkmark is set. However, the TOPMOST checkmark must not be set, as TOPMOST has priority over Min.
<b>Clear</b>	Deletes the content of the list box if the check mark is set.
<b>Stop</b>	Stops the entries into the list box if the check mark is set.
<b>Mouse Position</b>	Shows the mouse coordinates in the list box if the check mark is set.
<b>Buttons:</b>	
<b>Execute</b>	Executes the selected app start script.
<b>Apply</b>	Executes the selected window arrangement.
<b>SaveAs</b>	Save as. Saves the current window arrangement under the assigned file name. If no file name has been entered, a file name is automatically generated using SaveAs, date and time.
<b>Open Help</b>	Opens the help in the currently active language.
<b>Show MonRes</b>	Shows the monitor resolution.
<b>Start Arrangement</b>	Restores the start arrangement (when the SCM was started).
<b>Copy Size</b>	Copies the size of the last active window.
<b>Paste Size</b>	Applies the copied size to the last active window.
<b>Minimize All</b>	Minimize all windows.
<b>Maximize All</b>	Maximize all windows.
<b>Tiles All</b>	Show all windows as tiles, approx. 1/3 of the monitor width, approx. 1/3 of the monitor height per tile.
<b>Save SD</b>	Saves the current window arrangement as the session default.
<b>Apply SD</b>	Applies the saved session default.
<b>Eng/Ger</b>	English/ German language changeover.

## Screen Manager

### 3. Description of the built-in Commands

The following commands are available to the user:

Command	Description
1. Show the monitor resolution(s)	Show the monitor resolution(s) in the list box.
2. Show the size and position of all windows in the list box	Displays the sizes and position of all windows in the list box.
3. Show the size and position of the last active window	Displays the size and position of the last active window in the list box.
4. Copy the size of the last active window	Copies the size of the last active window.
5. Apply the copied size to all other windows	Applies the copied size to all other windows.
6. Apply the start size and position to the last active window	Applies the start size (at the time of SCM start) and position to the last active window.
7. Set the last active window to TOPMOST	Sets the last active window to TOPMOST.
8. Reset the last active window to no-TOPMOST	Resets the last active window to no-TOPMOST.
9. Move all invisible windows to the visible area	Moves all invisible windows to the visible area. This actually happens (about once a year for me) if you use the same settings file on different PCs with different monitor configurations.
Open the directory in Windows Explorer:	
10. Open the main directory of the application	Opens the main directory of the application in Windows Explorer.
11. Open the subdirectory _AppStartScripts	Opens the _AppStartScripts subdirectory in Windows Explorer.
12. Open the subdirectory _Arrangements	Opens the _Arrangements subdirectory in Windows Explorer.
13. Open the subdirectory _Exports	Opens the _Exports subdirectory in Windows Explorer.
14. Open the subdirectory _Help	Opens the _Help subdirectory in Windows Explorer.
15. Open the subdirectory _License	Opens the _License subdirectory in Windows Explorer.
16. Open the subdirectory _Settings	Open the _Settings subdirectory in Windows Explorer.
Open file in the standard text editor:	
17. Update the files in the _AppStartScripts directory	Updates the files in the _AppStartScripts directory.
18. Update the files in the _Arrangements directory	Updates the files in the _Arrangements directory.
19. Open the Settings.txt file	Open the Settings.txt file with the standard text editor.
20. Open the referenced AppStartScripts file	Opens the referenced AppStartScripts file with the standard text editor.
21. Open the referenced Arrangements file	Opens the referenced Arrangements file with the standard text editor.
Further commands:	
22. Show the size and position of all windows in a text file	Shows the sizes and positions of all windows in a generated and exported text file that is opened automatically.

## Screen Manager

Command	Description
<b>23. Show the size and position of the last window in a text file</b>	Shows the size and position of the last window in a text file that is opened automatically.
<b>24. Export the content of the list box to a text file</b>	Exports the content of the list box to a text file that is opened automatically.
<b>25. Delete all SaveAuto text files in the _Exports directory</b>	Deletes all SaveAuto text files in the _Exports directory.
<b>26. Delete all SaveAs text files in the _Arrangements directory</b>	Deletes all SaveAs text files in the _Arrangements directory.
<b>27. Apply the entered window width/height (e.g. 300x400) to the last active window</b>	Applies the entered window width/height (e.g., 300x400) to the last active window.
<b>28. Restore Screen Manager's original size</b>	Restores the Screen Manager to its original size (e.g., from the tile size).

### 4. Description of the User Settings

The Screen Manager saves the following settings in the \_Settings subdirectory when it is closed:

- The language to be used
- The window position to be applied
- The applicable state of the 'Topmost' checkbox
- The applicable state of the 'Minimize' checkbox
- The ExcludeMax string

The ExcludeMax string makes it possible to exclude certain applications from the maximize command. Certain applications (e.g., the SCM) have a fixed size, i.e., you cannot change their size by dragging the mouse. Unfortunately, Windows does not take this into account in the relevant API commands, i.e., such size changes are executed. The ExcludeMax string can be used to prevent these size changes.

The ExcludeMax string must be added manually. When executing relevant API commands, no full comparison is performed, only a Find (short name of the application in the full path of the EXE). You should therefore specify the shortest possible name without version number in the ExcludeMax string so that it still works in a new SCM version without adaptation. Applications to be excluded must be separated from each other by semicolons. A semicolon must also be placed after the last entry.

### 5. Revision List

Date	Version/Chapter	Description of the change
09.09.2024	v1.09 (b)	First complete SCM version.