ScreenSelect

by Markus Aalto

1 Introduction

ScreenSelect was programmed for users which commonly have two or more screens open at the same time. It shows a list of screen names in a listview gadget, so that user can then do a double click selection to get new screen to front. It is mainly designed to be used with public screens but it also works with normal screens if wanted so. Only requirement is KS2.04 (V37) or greater. It is implemented as a commodity and it also sports a font-independent interface. In addition ScreenSelect allows quick change of default public screen and easy binding of hotkey to any public screen.

2 Installation

Workbench users could just put ScreenSelect to WBStartup directory. DONOTWAIT-tooltype is already set in ScreenSelect's icon, so you can install it simply by moving its icon.

CLI users may want to start ScreenSelect from S:User-Startup, so you first have to make minor modifications to User-Startup-file and then copy ScreenSelect somewhere along the path (for example to C-directory).

I have also provided a program called SS_AutoUpdate (see Chapter 5 [AutoUpdate], page 5) which you can also drop to the WBStartup directory or start it from User-Startup. This file patches few system functions to enable automatic update of listview gadget and Default PubScreen text field.

3 Tooltypes and CLI options

ScreenSelect has normal commodity tooltypes and few other of its own.

3.1 Workbench Tooltypes

'CX_POPUP'

> This switch enables ScreenSelect to open its window when it's started.

'CX_PRIORITY=xxx'

> Where xxx is decimal number. Default is 0.

'`CX_POPKEY=HOTKEY`'

> Default is `shift esc`. For example you could use `CX_POPKEY=alt shift del` to change hotkey.

'`ALLSCREENS`'

> This switch makes `ScreenSelect` to list all screens. Normally only public screens are shown.

'`SMARTPOINTER`'

> This switch makes ScreenSelect to automatically put mousepointer over the previous selection in listview gadget.

'`QUICKKEYS=SCREENNAME_1=Hotkey_1|SCREENNAME_2=Hotkey_2|...`'

> With `QUICKKEYS` you can bind a hotkey to public screen. Note that screen name must be exact copy of the public screen name.

'`AUTOACTIVATE`'

> This switch enables a feature which tries to activate screen (actually it activates a window) automatically. AutoActivation procedure tries to be smart when desiding which window to activate. And it can only work if there isn't many windows open, so many times it does nothing. Here are checkings it makes.
>
> 1. Check if there is only one window open. And if this is the case then activate it.
> 2. Check if there are two windows open. If one is BACKDROP window then activate the other one. Otherwise do nothing.
> 3. If there are more windows open, do nothing.

'`SCREENKEY=HOTKEY`'

> If you set this one then `ScreenSelect` installs a hotkey, which you can use to set active screen as new default public screen. This makes use of public screens as replacement for `Workbench` screen very fast and easy.

3.2 CLI options

'`CX_POPUP`'

> This is a switch. See Section 3.1 [Workbench Tooltypes], page 1.

'`CX_PRIORITY=xxx`'

> Same as `Workbench` option.

'`CX_POPKEY="HOTKEY"`'

> Same as `Workbench` option, but remember to put '"' marks around `HOTKEY` name.

'`ALLSCREENS`'

> This is a switch.

'SMARTPOINTER'

> This is a switch.

'QUICKKEYS="SCREENNAME_1=Hotkey_1|SCREENNAME_2=Hotkey_2|..."'

> Same as `Workbench` option, but here you should put '"' marks around whole string.

'AUTOACTIVATE'

> This is a switch.

'SCREENKEY="HOTKEY"'

> Same as `Workbench` option, but put '"' marks around HOTKEY string.

3.3 Small example

Here is a small example of the tooltypes for `Workbench` users.

```
DONOTWAIT
ALLSCREENS
CX_POPKEY=numericpad (
QUICKKEYS=Workbench=numericpad 1|MEMACS_01=numericpad 2
SCREENKEY=lcommand p
```

These options enable `ALLSCREENS` option. Hotkey is bind to '(' key in numpad keys. 'Workbench' screen is bind to '1' key in numpad keys. 'MEMACS_01' screen is bind to '2' key in numpad keys. Public screen selection key is set to `left AMIGA p`.

4 Use

When you start `ScreenSelect` it normally disappears to background. You can activate it by making hotkey selection (default `shift esc`). When activated `ScreenSelect` opens a window with names of currently open screens. By double-clicking one of the names in listview-gadget you can quickly deactivate `ScreenSelect` and let it switch your selected screen to frontmost. However it won't normally activate it for you . It tries to do activation if AUTOACTIVATE option is ON, but it still may fail. So you usually have to click in one of the windows in that screen. If you have selected one of the names in listview-gadget but not double-clicked, then by pressing `Activate` gadget, `ScreenSelect` makes screen switching operation. By pressing `Cancel` or `WINDOWCLOSE` gadget you get back to the screen you were before you activated `ScreenSelect` with hotkey.

You can quit `ScreenSelect` either with `Quit` gadget, `Quit` menu item or `Commodity Exchange` program's `Kill` gadget.

Below the `Quit` gadget there is also a `Update` gadget which scans through screens in your system and updates listview. You can also toggle `ALLSCREENS`-flag by pressing cycle gadget you can found below `Update`-gadget. There should be reading `Public Screens` or `All Screens`. Listview gadget updates can also be automatic. For more info, please See Chapter 5 [AutoUpdate], page 5.

Below the listview is a text gadget containing default public screen's name. The update of this field is automatic if `AutoUpdate` patches have been installed. Otherwise this field may not always be correct.

There is also a menu attached to `ScreenSelect's` window. All selections have own keyboard shortcuts. Note that although `Activate` and `Cancel` gadgets don't have respective menu items, you still have keyboard shortcuts. By pressing `A` from the keyboard you select `Activate` gadget and with `C` you can select `Cancel` gadget.

New `SMARTPOINTER` option makes bounching between two screens very fast and easy. It moves system mousepointer to the place of previous screen's name in a listview-gadget. You can simply double-click to get back to the previous screen. For example if you have three screens in you system. One is `Workbench`, one is your editor's screen and one is your communication program. First you would be in `Workbench` screen and activating `ScreenSelect`. Select editor screen and activate it. Then activate `ScreenSelect` again and now mousepointer is placed over `Workbench` name. Now if you double-click you get back to `Workbench`. Do same procedure again and now pointer is placed over your editor screen's name.

Even if `SMARTPOINTER` option isn't ON you still can have almost as easy way to bounch between two screens. When you activate `ScreenSelect` window, previous screen name is automatically in a selection field of listview-gadget. Now if you press `A` or click `Activate` gadget you can get back to previous screen. Easy!

Now that there are lots of programs supporting hotkeys and public screens there is a great demand for easy selection of system's default public screen. For example if you are using an editor and want to open a new program to editor's screen so that you can continue editing, you can set new default public screen with hotkey (Check SCREENKEY tooltype) and after that all programs supporting default public screen open their windows to editor screen.

5 AutoUpdate

AutoUpdate (Real name is SS_AutoUpdate which stands for ScreenSelect AutoUpdate) in-
stalls a small patch to Intuition Library's OpenScreen, OpenScreenTagList, CloseScreen and
SetDefaultPubScreen functions. These patches allow ScreenSelect to automatically update
listview gadget when new screens are opened or old ones are closed. This also enables automatic
update in Default PubScreen field. However you don't have to install these patches to be able to
use ScreenSelect. They are completely optional.

Patches are installed by running SS_AutoAdjust[1] program from Workbench or CLI. After you
have made installation, you can't remove them. All four patches take about 100 bytes of free ram.

6 Misc

This is a common procedure ScreenSelect makes when it hears about hotkey selection.

```
if hotkey pressed then
    if active screen is public screen then
        lock it and open ScreenSelect window into it.
    else
        lock default public screen and open window to it.
```

So this means that ScreenSelect window isn't opened to foreign applications screen. It's
only opened to public screens. This may annoy some users but I made this decision to make
ScreenSelect more system friendly. Remember that if you select new frontmost screen then there
really isn't much harm done. And even if you just press Cancel-gadget you get always back to
your old frontmost screen.

ScreenSelect also tries to make sure that all screen pointers passed back to OS, are always
valid. This is easy with public screens but requires little more work with normal screens. I hope I
got it right!

Also there are programs which don't give name for a new screen. Screens like that are shown
in listview as (Nameless screen). I hope that programmers would always specify a name for a
screen, but I guess that sometimes it just isn't possible.

---

[1] Users who don't trust patches made by another programmer, I have included the source code
for SS_AutoAdjust. You can compile it yourself if you like.

Appendix A History

A.1 Version 1.0

First release 07-Sep-1992.

A.2 Version 1.1

This version was not released. Compilation date was 14-Sep-1992.

- Changed `Workbench` tooltypes `CX_POPUP` and `ALLSCREENS` to simple switches. They are normally OFF, but if you specify one of them then that option comes ON.
- Added `SMARTPOINTER` option. When this one is active `ScreenSelect` always tries to position mouse pointer over previous screen selection. This makes it very easy to bounch between two screens.

  [Suggested by Sfefan Boberg]
- Added `QUICKKEYS`. Now you can bind own hotkey to every public screen. This should be used only with public screens. Although it might work with all screens I don't support it.

  [Suggested by Stefan Boberg]
- Now if `ScreenSelect's` window is active (open) when hotkey is pressed then that window's screen is brought to front.
- Added menus to `ScreenSelect's` window. Now most of the features are selectable with Menu command keys.
- Added `About` window to show some information.
- Now previous screen's name is shown in listview gadgets selection area automatically. This makes screen bounching very easy as `SMARTPOINTER` option does.
- Now `Activate` and `Cancel` gadgets have keyboard shortcuts.

  [Keyboard shortcuts were suggested by David Corn]

A.3 Version 1.1a

Compilation date was 23-Sep-1992.

- Fixed a bug which happened when `SMARTPOINTER` option was ON and screen font was too big to be used in `ScreenSelect's` window and we had to use topaz80. This caused pointer to be moved to wrong position.

A.4 Version 1.1b

Compilation date was 13-Oct-1992.

- Just changed a initial window height from 94 pixels to 96. Now normal topaz80 font shouldn't cause window resizing when only Workbench (or screen names of similar lenght) is in a list.

A.5 Version 1.2

Compilation date was 07-Dec-1992.

- Previously `ScreenSelect` exited when it couldn't find screens. This made it impossible to put it in `Startup-Sequence` or `User-Startup` file before `Workbench` screen is opened.
  [Reported by Andreas M. Kirchwitz]
- From now on, version number format should be Amiga Style Guide compliant.
- Added `AUTOACTIVATE` option.
- Added `SCREENKEY` option which allows easy default public screen selection with hotkey.
- Added `Default PubScreen` text gadget.
- First version compiled with SAS/C V6.1. Size reduced almost 2k compared to V1.1b.

Appendix B Contents of ScreenSelect distribution

Here are listed all files and directories in current version 1.2 distribution. These files should always be present when `ScreenSelect` is distributed.

```
ScreenSelect (dir)
    AutoUpdate (dir)
        SS_AutoUpdate

        SS_AutoUpdate.info

        SS_AutoUpdate.asm

        SS_AutoUpdate.asm.info

    Docs (dir)
        ScreenSelect.doc

        ScreenSelect.doc.info

        ScreenSelect.dvi

        ScreenSelect.dvi.info

        ScreenSelect.guide

        ScreenSelect.guide.info

    AutoUpdate.info

    Docs.info

    ScreenSelect

    ScreenSelect.info

    IMPORTANT

    IMPORTANT.info

ScreenSelect.info
```

Appendix C Contact address

For suggestions, bug reports etc... You can reach me at

- Internet

    s37732v@vipunen.hut.fi
- SnailMail

    Markus Aalto, Muurahaisentie 11a, 01490 VANTAA, FINLAND

Table of Contents