Ursula Screensaver
Functional Specification

COMPANY CONFIDENTIAL

Document Ref:	1309,215/FS
Project:	Ursula
Revision:	D*** LIVE ***
Date:	29th April 1998
Author(s):	Richard Leggett
Change:	None

Contents:

1.0 Overview

Previous versions of RISC OS included support for automatic blanking of the screen after a configurable period of inactivity; however, no support was provided for more interesting screensavers. The Ursula release of RISC OS aims to remedy the situation by providing a new version of the *Screenblanker* module, along with a new module called *ScrSaver*.

The new modules enable an ordinary WIMP application to be run when the configured period of inactivity occurs and provide a mechanism for letting the application know when the screen needs to be restored.

2.0 Outstanding Issues

None.

3.0 Technical Background

3.1 ScreenBlanker and ScrSaver

N/A

3.2 Screensaver applications

Whilst it would be possible to write single-tasking screensaver applications, they should be multi-tasking, so that other applications can continue processing in the background.

Upon startup, a screensaver is passed the address of a pollword as the first command line argument. The application will pass this value to Wimp_Poll and install a PollWord_NonZero event handler to monitor it. The first call to Wimp_Poll should have the high priority pollword bit set (bit 23) but subsequent calls should not have this bit set. Setting it for the first Wimp_Poll ensures that there won't be unnecessary screen redraw if the user causes an unblank only shortly after a blank.

If the Pollword_NonZero event occurs, the screensaver should exit immediately. In doing so, its window will be removed and the screen will be restored.

A handler for the Pointer_Entering_Window event should also be installed. When this event occurs, the application should switch the pointer off. The program exit code should then switch the pointer on again (to shape 1) before finishing.

The application blanks the screen by opening a black or transparent window over the whole of the screen and then using null events to paint to it. Upon startup, the application should read details about the current screen mode and use this to decide how large to make the window.To ensure that no other windows open over the top of the screensaver window, the Ursula window manager provides the facility for windows to be flagged as foreground windows. These windows can only be obscured by Wimp error boxes.

[To mark a foreground window, windows should be opened with a handle-to-open-behind of -1 and with bit 23 of window flags set. For full details, refer to the Ursula Window Manager Changes Functional Specification [1].]

The !Run file for a screensaver application should include the line:

```
Set ScrSaver$Running Yes
```

and its exit routine should unset the system variable.

In order to avoid having to register application names, screensavers should:

- Not have any application sprites (ie. !Sprites22 file)
- Not use a system variable of the form `Appname$Dir` but instead use the reserved variable `BlankTask$Dir` to point to it's application directory.

4.0 User Interface

Screensaver applications are kept hidden away within the !Boot.Resources.Configure.ScrSavers directory. The !Configure application[2] allows configuration of the screensaver and it produces a list of available savers by scanning this directory. Configure is responsible for writing an Obey file which will be run on startup and which sets the value of `ScrSaver$Dir` and `ScrSaver$MinMem`. The latter variable is used by *ScrSaver* to decide if there is sufficient memory to run a given screensaver. Configure decides on a value for it by looking at the -min switch of the WimpSlot command in the application's !Run file.

Each screensaver may optionally have its own configuration window and this should be provided by an application within the screensaver's own application directory. The application should be called `!SvrSetup` and should expect command line arguments which tell it where to open the top-left corner of its window. These will be of the form:

```
!SaverName.!SvrSetup -openat 364 822
```

The window should provide Configure style Cancel, Set and Default buttons and should keep any choices in a file within the `<Choices$Write>.ScrSavers` directory, rather than within its own application directory (as this can create problems on networked systems).

5.0 Programming Interface

N/A.

6.0 Data Interchange

N/A.

7.0 Data Formats

No new data formats are involved.

8.0 External Dependencies

The ScrSaver module uses a new SWI which is provided in the Ursula release of the Switcher.

A new version of !Configure will allow easy selection of favoured screen saver.

9.0 Acceptance Test

N/A.

10.0 Non Compliances

None.

11.0 Development Test Strategy

N/A.

12.0 Product Organisation

The new version of *ScreenBlanker* and the new module *ScrSaver* will be built into the Ursula ROMs. An as yet undetermined, number of screen saver applications will be included in the Ursula disc build.

13.0 Future Enhancements

None planned.

14.0 Glossary

N/A

15.0 References

1. 1309,213/FS Ursula Window Manager Changes
2. 1309,214/FS Ursula !Configure Changes

16.0 History

N/A.