Ursula Pinboard Changes
Functional Specification

COMPANY CONFIDENTIAL

Document Ref:	1309,210/FS
Project:	Ursula
Revision:	F *** PROVISIONAL AND SUBJECT TO CHANGE WITHOUT NOTICE ***
Date:	28th January 1998
Author(s):	Richard Leggett
Change:	Updated User Interface, Data Interchange.

Contents:

1.0 Overview

This document covers the changes to the Pinboard for the Ursula release of RISC OS. The 'old' Pinboard, supplied with RISC OS 3.71 behaves in an inconsistent manner, treating file icons (aliases) and iconised windows in a very different and potentially confusing way. The Ursula release ('new' Pinboard) will hopefully remove some of these inconsistencies, as well as adding extra functionality, specifically:

- Options to iconise windows to a corner of the backdrop.
- Options to tidy icons to a corner of the backdrop.
- A Filer style menu, providing operations on selections (of files and window icons).
- The ability to have JPEG images as backdrops.
- The ability to set the background colour of the desktop.

In addition, the configuration of the Pinboard will be moved to the Configure application and the menu structure tidied up.

2.0 Outstanding Issues

None.

3.0 Technical Background

3.1 Inconsistency

The Pinboard treats file alias icons and iconised window icons in two different ways and maintains a separate linked list for each. In both cases, the list structure contains details about things such as the icon handle, x and y co-ordinate, spritename etc., but other details are different.

Having two different types of icon on the Pinboard can cause confusion. For example, when the user performs a 'Tidy' operation, this operates on only the file alias linked list, moving through the list and arranging icons in the top left hand corner. It doesn't check the iconised window list to see if any window icons are going to be obscured and it doesn't move window icons in any way.

3.2 Iconising behaviour

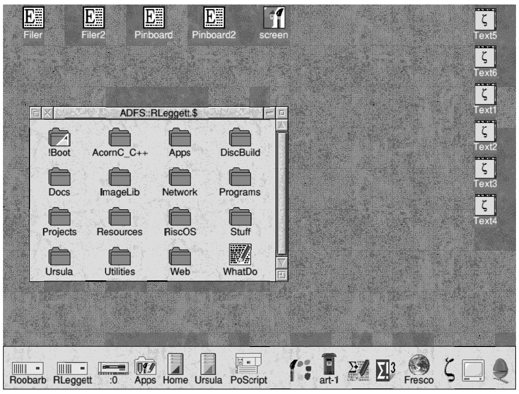
When the user Shift+Clicks on the close icon of a window, the Window Manager sends out a Message_Iconize which is typically claimed by the Pinboard application. The Pinboard acknowledges the message and then sends a Message_WindowInfo to the window.The window can choose to reply to the message if it wants to help with the iconising procedure. The result of all these messages is an icon appearing on the Pinboard centred on where the close icon of the window was.

With the latest version of the Window Manager, an iconise button is present next to the toggle size icon. Clicking on this follows the same iconising behaviour as a Shift+Click and so the iconised window again appears centred on where the close icon was. This is clearly confusing to the user.

4.0 User Interface

4.1 Window and file iconising

The Pinboard will be changed to provide an option to send icons to a corner of the Pinboard when a user iconises a window, and to stack them horizontally or vertically if space is already occupied. Logically, because the iconise icon is in the top right of a window, the default action will be to send iconised windows to the top right and to stack downwards, ie. :



This behaviour will only apply when the user clicks on the iconise button. However, if they Shift+Click on the close button, the window will be iconised underneath the close icon, as has always been the case. An option will also be provided to iconise 'at button', so that a click on the iconise button would result in the window's icon being placed underneath that button (in a similar manner to a Shift+Click on the close button).

An option will also be included to use the IconiseToIconBar option which has always been present in Pinboard, but not documented in manuals or in command line syntax.

A new option will allow selection of the corner that files are tidied to (when the user chooses 'Tidy' from the Pinboard menu), plus the direction of their stacking.

4.2 Consistency in behaviour

Selections will now apply to both windows and files and the 'Select All' option will be modified to lead to an optional menu offering 'Select files' and 'Select windows' options. The 'Remove icon/Selection' option will go, to be replaced by a 'Selection' submenu. This submenu will have options to Tidy the selection, to Open (run each file or open each window) and Remove selections. The Remove option is only applicable for files, so if any window icons are in the selection, the option will be greyed out.

Choosing Tidy (after having selected file icons and/or window icons) will result in selected windows being sent to one corner (the same that they are iconised to) and selected files to another (also configurable by the user). Windows will be dealt with first, followed by files. The files will avoid overlapping windows by moving on to the next free grid position if the current one is occupied. The order in which windows and icons are stacked will be the order in which they were placed on the backdrop, as this is the order in which the linked list is stored.

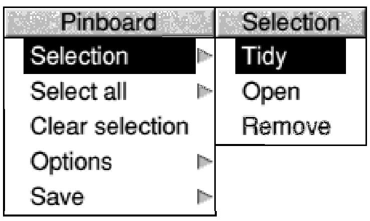
Additionally, if iconised window icons are sent to a corner, they will avoid obscuring any already present file or window icons, by moving onto the next free grid position.

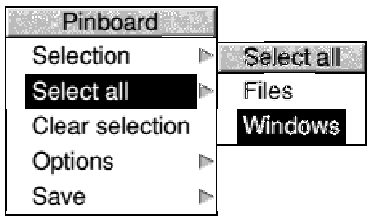
4.3 Menu changes

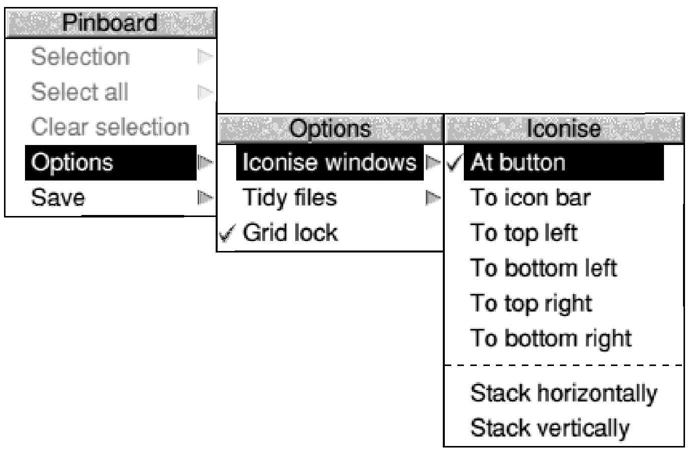
All Pinboard options will now be set from within the Configure application. However, the Pinboard menu will still provide selection of the options which will apply only while the current Pinboard session is in progress (ie. they will not be saved and will be lost on shutdown).

A click on 'Select all' will now select all file and window icons. There will also be a submenu which allows selection of just files or just windows. The make/remove backdrop options are to be taken out and backdrop control left to Configure. Finally, the 'Info' option will be removed from the Pinboard menu to make it appear a more integral part of the OS.

So, the new menu will have the following appearance:







The 'Save' submenu still leads to a Save dialogue box, but the default filename for this will be <Choices$Write>.Boot.Tasks.Pinboard meaning that if the Pinboard is saved to this filename, it will automatically be restored on boot-up. Additionally, a click on the 'Save' menu option (rather than in the save box) will save to the current filename.

4.4 Backdrops

The backdrop mechanism will be changed to allow tiling of JPEG images. To the user, the mechanism of putting images on the backdrop will be the same as for sprites. However, JPEG images will be plotted with the RISC OS JPEG module, rather than SpriteExtend, and as a result may be slow. Despite this, it appears to be a feature that users want.

Should the user decide not to use an image for the backdrop, a Configure option will allow selection of the colour of the backdrop, which is currently always grey.

4.5 Default save filename

The default save filename will be changed to <Choices$Write>.Boot.Tasks.Pinboard. If saved with this filename, then the Pinboard will automatically be restored on boot. Additionally, a click on the Save option in the Pinboard menu (rather than opening the save box) will save the Pinboard using the last filename.

4.6 Command line interface

Because of all the options described above, a new *command is provided to set them. Typically this will be written to a file by Configure, which will then be executed on machine boot. It's syntax is:

```
PinboardOptions [-Grid]
                [-IconiseTo<Iconbar|TopLeft|BottomLeft|TopRight|BottomRight>]
                [-IconiseStackVertical]
                [-TidyTo<TopLeft|BottomLeft|TopRight|BottomRight>]
                [-TidyStackVertical]
```

- If the 'IconiseStackVertical' switch is omitted, horizontal stacking of window icons is assumed.
- If the 'FilesStackVertical' switch is omitted, horizontal stacking of file icons (when tidying) is assumed.
- If the 'IconiseTo...' switch is omitted, windows are iconised to the pointer.
- If the 'TidyTo...' switch is omitted, files are tidied to the top-left.

The Backdrop command will be changed to allow setting of the backdrop colour. It's new syntax will be:

```
Backdrop [-Colour <&GGBBRR00>]
         [-Centre | -Tile | -Scale | -Remove] [<pathname>]
```

If the Colour switch is omitted, then the background is set to the grey (&77777700) used in previous versions.

5.0 Programming Interface

None.

6.0 Data Interchange

The WIMP will be changed to broadcast a new message, Message_IconizeAt, just before it broadcasts Message_Iconize. This new message gives details about how and where the iconising operation was performed. It's format is detailed below:

Message number &400D0   Message_IconizeAt

+20	window handle of window being iconised.
+24	task handle of window being iconised.
+28	x co-ordinate to iconise at.
+32	y co-ordinate to iconise at.
+36	Flags - bit 0 set means window was iconised with a shift + click on close button.

It's use is twofold. Firstly, a shift-click on the close icon will always iconise a window beneath the icon, so the message enables Pinboard to tell that this was the case. Secondly, when the user has selected the 'iconise at button' behaviour, Pinboard centres the window's icon at the co-ordinates specified in the block above.

In the absence of a Message_IconizeAt, Pinboard will revert to it's old behaviour of positioning window icons beneath the window's close button.

The message may also be used by other applications that wish to iconise windows.

7.0 Data Formats

No new data formats are involved.

8.0 External Dependencies

A new version of the Window Manager, which generates Message_IconizeAt will be required. It's absence will not cause Pinboard to fail, but will mean icons will be iconised to the close button, even when the user clicks on the iconise button.

A new version of Configure is required which will allow Pinboard options to be set.

9.0 Acceptance Test

Key areas for acceptance of the new Pinboard are :

- Reliability - the new Pinboard should aim to fix some of the bugs currently present and not to create any new ones.
- Speed - the new Pinboard will have to do a lot more processing when icons are tidied or sent to a corner. All operations should be no slower than the previous version running on the previous hardware. Some typical tests are:
    - Place 50 file icons and 50 window icons on the backdrop, then Tidy all.
    - Put more icons on the backdrop than would fit if placed at grid spacings, then Tidy all.

- Place 50 file icons, in a corner then iconise 10 windows to the same corner.

- Memory usage - the new module should be no more than 16k larger than the previous Pinboard and should require no more than 4k extra workspace.

10.0 Non Compliances

N/A

11.0 Development Test Strategy

Once reliability and consistency have been obtained, then volunteers within Acorn will be sought to use the new Pinboard on a day-to-day basis on their own machines. Any less obvious problems should show up during this stage.

12.0 Product Organisation

The new version of Pinboard is intended to be built into the Ursula ROMs.

13.0 Future Enhancements

This will be filled in from Outstanding Issues at a later date.

14.0 Glossary

N/A

15.0 References

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

16.0 History

Revision	Who	Date	Comment
A	RML	03-06-1997	Started
B	RML	08-07-1997	Updated

C	RML	24-07-1997	Updated after spec review
D	RML	12-08-1997	Released for further comments
E	RML	20-11-1997	Changed command line interface.
F	RML	28-01-1998	Updated to keep in line with implementation. Revision number 0.04 renamed to E.