Black Draw Functional Specification

```
-----------------------------------------
| Drawing No : 1303/010/FS/3P95         |
|      Issue : 1/3P95                   |
|       Date : 27/07/94                 |
|     Author : Jonathan Coxhead         |
|     Sheets :                          |
|  Last Issue: None                     |
-----------------------------------------
```

History

1.00 MRC 21/3/95 Updated for Developer pack.

Contents

Background

It is desirable that Draw should track the new faciities introduced into the operating system. It is already capable of handling the new sprite formats introduced under Black. Draw will be extended to support a new JPEG object type.

User Interface

JPEG's may be dragged around the document and resized using the scaling "ear" or the various scaling menu entries (Transform => x scale, Magnify etc). However, any attempt to rotate a JPEG object will leave it unaffected in orientation (though it will be moved, if necessary). This is because the underlying JPEG rendering engine in Sprite Extend does not support such an operation. See appendix 1.

In order to reduce potential confusion over this, the rotate "ear" will be removed from objects that do not rotate. These are text lines in system font, text areas, JPEG objects (currently) and groups containing any of these. Since Sprite Extend supplies an interface to determine whether JPEG's can be rotated, this will be used at run time so that if an upgraded SpriteExtend supporting JPEG rotation is loaded, Draw should then support JPEG rotation itself and leave the rotate "ear" in place for JPEG objects.

To allow the re-exporting of a JPEG from a file that contains one an extra menu entry ('Photo') will be provided on the Save menu. This is only available when the selection includes a single JPEG. The menu entry will have help text

Move the pointer right to get a save box for the selected photo

and the save-as dialogue box will have the standard Draw help text

Either drag SELECT from the file icon to a directory display, or fill in the name of a file and click SELECT on OK, to save as a photo.

The name of the file in which the JPEG is stored will be Photograph.

Data formats

To handle JPEG files in Draw, the Draw file type will be extended to understand a new object type, number 16, defined as

```
struct drawfile_jpeg
   {  os_box bbox;
      os_trfm trfm;
      int width, height;
      int len;
      byte data [UNKNOWN];
   };
```

(where this appears in the union type

```
struct drawfile_object
   {  int type;
      int size;
      union
      {  ...;
         drawfile_jpeg jpeg;
         ...;
      }
      data;
   };)
```

Like the transformed sprite object type, the transformation matrix is used to determine the scaling factors with which the image must be plotted. The bounding box is calculated from this. When the object is loaded into the draw file, its extent is read (using jpeg_info()), and these values are converted to OS units (making the assumption that 1 pixel is 2 OSU in the absence of information to the contrary) and stored in the width and height fields of the object. This is an optimisation, really: the values do not change, so they may be safely kept.

The representation inside Draw is just the original JPEG file, held in the structure, preceded by its length in bytes. The file is not converted to a sprite, but rather rendered directly when necessary. This makes the code much simpler, as no special action need be taken on a mode or palette change. Furthermore, memory requirements are minimised, and the image will be able to take full advantage of any increased resolution after a mode change.

A transformation matrix is provided in the JPEG object. This means that in principal the object can be rotated. This feature will be disabled initially, while Draw detects at run time that SpriteExtend does not support JPEG rotation.

Draw provides no editing capabilites for JPEG's (as for sprites), so the width and height are fixed values. The bounding box and transformation are initially calculated from these, but may subsequently be dragged by the user. When a JPEG is to be plotted, the current mode variables (XEIG, YEIG) and the transformation of the JPEG are used to calculate a plot position and scaling factors to be passed to jpeg_put_scaled().

Test strategy

Very large, very small and invalid JPEG files will be imported into Draw, and moved around and rescaled. The screen mode will be changed to all the available possibilities, and the image checked visually to ensure that the results are appropriate to the facilities available (e g, good contrast in grey scale modes, etc).

The export code will also be tested, with various combinations of sprites/JPEG's selected. A binary comparison made between the original JPEG and the exported JPEG (they must be identical).

The output from a Draw file to a printer will be checked, and must be at least as good as that produced by printing the sprite as converted by ChangeFSI or Paint.

Acceptance tests

The version of Draw produced by this work should be no slower than the Medusa release in all areas that are shared, i e, in the absence of JPEG objects. The size of the new code should be less than 10K squeezed.

The code for performing the rotation if such a Sprite Extend is present cannot be tested until such time that the support is actually provided.

External dependencies

None of the above can be tested properly until the Sprite Extend module is enhanced to provide the JPEG SWI's. A lot of the user-interface code can be written before then, however, with the proviso that JPEG's will appear as blank rectangles if the module is not present.

Testing the output quality of the new objects will have to await support from the printer driver modules.