



# PrivateEye

Image Viewer  
by David Thomas, © 1999–2012  
version 2.99 (05 Feb 2012)

## Welcome

PrivateEye is an image viewer for RISC OS.

It requires RISC OS 3.6 or later and a Boot sequence with the Nested Wimp and a 32-bit Shared C Library.

## Features

- Loads and displays bitmap and vector images
  - Sprites, JPEGs, GIFs and PNGs
  - DrawFiles and ArtWorks
- Bitmap effects
  - Adjust gamma, brightness and contrast
  - Blur and sharpen
  - Change saturation and apply histogram effects
- Alpha channel support
- Bitmap rotation with interactive preview
  - Rotation is lossless for JPEGs
- Native JPEG display using SpriteExtend
  - Inbuilt lossless “cleaning” transparently loads progressive JPEGs
- JFIF, Exif and Adobe metadata information display
- Multiple-channel histogram
- Display images may be saved
  - Convert JPEGs, GIFs and PNGs to into Sprites
- Any number of images may be open concurrently
- Customisable key map
- Lots of interactive help (use it!)

# Supported Image Formats

PrivateEye converts images as necessary into a RISC OS-native format. This means that GIFs, PNGs and (optionally) JPEGs are converted into Sprite format when they are loaded.

The converted image is referred to as the *display image*. Operations such as saving, rotation and the effects system operate on the display image only.

## Sprites

All varieties of Sprite are loaded and displayed so long as the OS understands them. For example, RISC OS 5 will be unable to load the alpha-blended and CMYK format Sprites native to RISC OS Select.

16 million colour format Sprites are checked for alpha channel data and rendered with transparency if found. The alpha data is expected to reside in the top byte of each 32-bit pixel. This is the same format used by *Tinct*, *Composition* and *Variations* but different to the format used by RISC OS Select which stores the alpha in the mask data.

Multiple Sprite files will only show the first Sprite in the file.

## JPEGs

RISC OS's SpriteExtend module, version 0.99 or later, is used to directly display JPEGs. This allows images larger than available free memory to be displayed by decompressing and plotting on the fly.

Progressive (multiple scan) JPEGs are supported. PrivateEye has an inbuilt version of *jpegtran* which automatically converts JPEGs into a baseline format that SpriteExtend can render.

JPEGs may be optionally be converted to Sprite when loaded (see the Choices window.)

## GIFs

GIFs are converted into Sprite format when loaded.

Animated GIFs will only show the first frame.

## PNGs

PNGs are converted into Sprite format when loaded.

Where possible, simple on/off transparency is retained as a Sprite mask. If the transparency is not simple then a 32bpp image with an alpha mask is created.

PNGs with 16 bits per component will be truncated to 8 bits, as there is no RISC OS format capable of holding that level of detail.

Where present, the *bKND* chunk is used to set the colour of the window background.

## DrawFiles

The DrawFile module is used to display DrawFiles.

A border of 16 OS units is added in display.

## ArtWorks

The ArtWorks rendering modules are used to display ArtWorks files.

AWViewer must have been seen by the Filer for this to work. Note that ArtWorks itself won't do: it must be AWViewer. All RISC OS machines since the Risc PC have shipped with a copy of AWViewer on the hard disc.

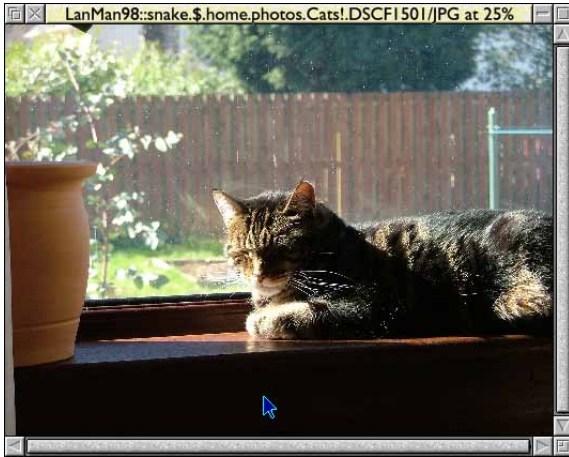
## Unrecognised Formats

If the file type is not recognised PrivateEye will inspect the contents of the file to see if it is a format that is understood. If so, it will set the file type appropriately and then load the file.

Otherwise, it will use Computer Concepts' FFG protocol to load images it does not natively understand. For this to work you must have *TransFSI*, *TranTIFF+*, or similar, installed.

# Viewer Windows

The *viewer window* is PrivateEye's main interface.



## Loading Images

To view an image, drag it from a directory display to the icon bar icon. You can load any number of images, each will be shown in its own viewer window.

To re-use an existing viewer window drag the image into that viewer window.

If the image does not have the correct file type, then

PrivateEye will attempt to identify the correct type and set it.

## Input Focus

When you click in a viewer window it gains the input focus. You can then use the keys detailed below to scroll, scale, etc.

## Scrolling Around

If parts of the image are not visible, you can scroll around either by dragging within the window using SELECT—the *grab* tool—or by using the cursor keys.

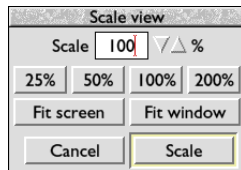
On suitable OS versions, the mouse scroll wheel may also be used to scroll vertically.

You can scroll to a specific point by holding CTRL and clicking SELECT at the desired position.

## Scaling The View

You may need to scale the image to fit it to the screen, or to view an area in detail. There are a number of controls for this.

The *Scale view* dialogue is available from the viewer menu or by pressing F11. The default scale of 100% shows the image at a 1:1 pixel ratio.



There are standard preset scales of 25%, 50%, 100% and 200%. **FIT SCREEN** fits the view to the size of the screen. **FIT WINDOW** fits the image to the current size of the viewer window. Click **SCALE** to make the scale level take effect.

With the Scale view dialogue open, you can also use the UP and DOWN keys to change the scale by 5%. This takes immediate effect. Using SHIFT in conjunction doubles the step size to 10%. (PAGE UP and PAGE DOWN perform the same function.)

### Scaling with the Keyboard

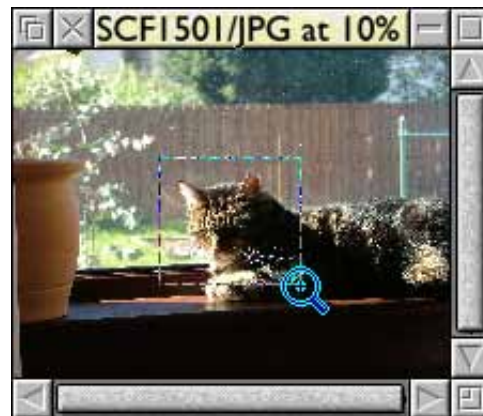
CTRL W and CTRL Q zoom in and zoom out respectively.

CTRL T toggles between the previously selected and the current scale level. CTRL D resets the scale to 100%.

### Scaling with the Mouse

A SHIFT SELECT click will zoom in around the clicked point. SHIFT ADJUST click does the same but zooms out. If multiple steps are configured then a zoom effect will be used.

Dragging ADJUST in the viewer window drags out a zoom box. The area of the image highlighted will be scaled to fit the window.



## Moving Between Files

Pressing SPACE, or alternatively PAGE DOWN, will move the viewer onto the next file in the directory. PAGE UP moves to the previous file.

The order of the files in the directory determines the previous / next viewed image.

## Saving

F3 opens the Save dialogue. This saves a copy of the display image.

## Delete

CTRL K deletes the currently viewed image. It uses the multitasking Filer\_Action window to perform the deletion so will interface with *Recyclone*, etc.

## New views

Should you need to view an image at two different scales, or view multiple different parts of a single large image, you can create a new view by pressing CTRL N. This opens up a new view of the same image.

Additional views require no significant additional memory.

## Embedding

Dragging with SHIFT CTRL SELECT inside the viewer window allows you to embed the viewer window inside another window.

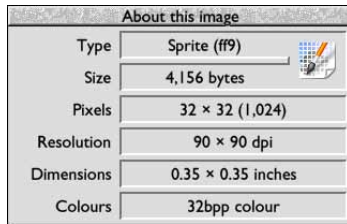
To un-embed a window perform the same action again but release the drag when over the icon bar.

This uses a feature of the Nested Wimp intended for embedding Java plugins inside web browser windows.

# Image Information

PrivateEye retains information about both source and display images, so there are **two** information windows.

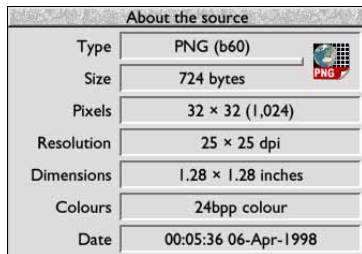
*Image Info* shows information on the display image. It is available from the viewer menu, or with SHIFT F1 and looks like this:



Notice that source info has an extra **DATE** field.

The information windows may often display the same information. However, if a change is made (e.g. a rotation) or a change in the image was involved when loading it (e.g. converting from JPEG to Sprite) then the information may differ. For example, Sprite files appear to be 4 bytes longer when loaded than when on disc. This is because the in-memory structure of a sprite is 4 bytes longer.

*Source Info* shows information on the source image. It is available from the viewer menu, or with CTRL F1 and looks like this:



# Metadata

Some image files may contain embedded information additional to the image data. PrivateEye calls this *metadata*. The *Metadata window* lets you see this information.

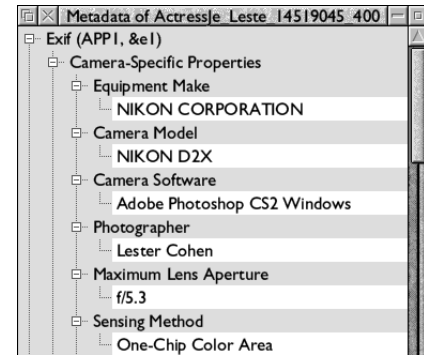
*The Metadata window is presently only available for JPEG images loaded in their native format. If you want to use this feature you may need to disable **CONVERT TO SPRITE** in the JPEG choices.*

Open the Metadata window by choosing **METADATA** from the viewer window's menu or by pressing SHIFT CTRL F1. One metadata window may be opened for each image.

The menu entry will be greyed out if no metadata is present.

## Use

The metadata is presented in a tree format like this:



Individual branches of the tree can be collapsed or expanded to hide or reveal relevant data by clicking the controls at the left.

Click MENU to open up a small menu which lets you collapse or expand all tree branches.



# Classes of Metadata

There are three common types of JPEG metadata which PrivateEye understands:

## Exif

- The standard for digital camera data.
- Contains information including time, date, camera make, model, resolution, focus and flash.

## Adobe

- Output by Adobe applications, primarily Adobe Photoshop.
- Contains information proprietary to Adobe applications such as effect settings.

The Adobe segment also includes *IPTC-NAA* data.

- Contains information about captions, headlines, keywords, photographer, copyright holder, etc.

## JFIF Comments

- JFIF provides for multiple blocks of plain, unformatted text data.

# Examples

## IPTC-NAA

Images containing this type of metadata may often be found on The Internet Movie Database

<http://www.imdb.com/>.

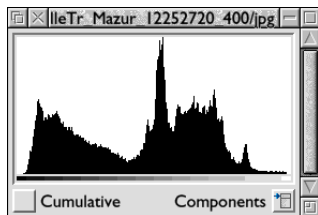
# Histogram

The *Histogram window* displays a histogram of an image's *luma*: the brightness portion of the image.

*The histogram window is only available for 8bpp grey and 32bpp colour bitmap formats. If you want to use this feature with a JPEG then **CONVERT TO SPRITE**.*

It can also show the histogram of red, green, blue and, where available, alpha, components.

Open the Histogram window by choosing **HISTOGRAM** from the viewer window's menu or by pressing CTRL I. One histogram window may be opened for each image.



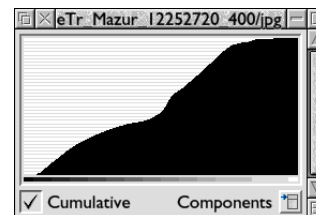
The histogram lets you see which tones contain the greatest amount of detail. In the above example, the low level at the right hand side of the graph indicates an absence of detail in the brightest part of the image.

You can use this information to guide your choices when

making adjustments with the Effects window.

## Display Options

The **CUMULATIVE** option displays the histogram as an increasing amount from left to right.



The **COMPONENTS** pop-up opens a menu which lets you choose which colour channel to view: luma, red, green, blue or alpha (where present).

## Scale Bars

Light grey horizontal bars are drawn across the display for every 3.125% of total pixels shown (32 intervals.)

## Luma Calculation

Luma is a weighted sum of gamma corrected components.

Luma here is calculated using the Rec. 601 coefficients of

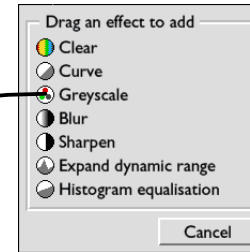
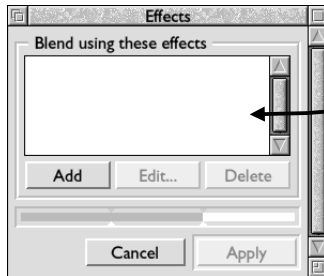
$$Y' = 0.299 R' + 0.587 G' + 0.114 B'.$$

# Effects

The *Effects* window lets you apply a multitude of effects to bitmap images.

*The Effects window is only available for 32bpp colour bitmap formats. If you want to use this feature with a JPEG then **CONVERT TO SPRITE**.*

Open the Effects window by choosing **EFFECTS...** from the viewer window's menu or by pressing CTRL E. An effects dialogue will appear containing an empty list:



Drag and drop an effect from the effects palette to the effects window to add it. The viewer window will immediately update with a preview of the effect.

The overall level of the effect is controlled with the slider at the bottom of the window. Dragging the slider updates the preview.

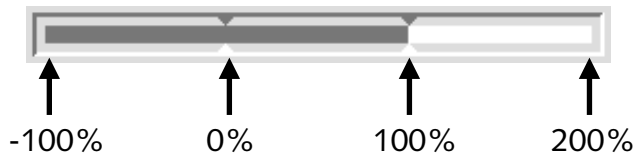
Click **APPLY** to make the effect permanent.

## Adding Effects

Click **ADD** to start adding effects. The *effects palette* window will pop up:

## Level Slider

Effects are computed and blended against the original image. The *level slider* controls the amount of blending.



Rather than ranging from 0% (original image) to 100% (processed image) it ranges -100%..200%. This is because the blender is *extrapolative*: it can produce results outside the normal range.

Setting the level outside the 0..100% range causes extrapolative results. For example, adding a “clear to black” effect then applying a level of -100% causes the image to be **brightened**.

## Multiple Effects

You can add multiple effects at once. Multiple effects are applied in sequence from top-to-bottom. The result is then blended with the original image.

## Editing Effects

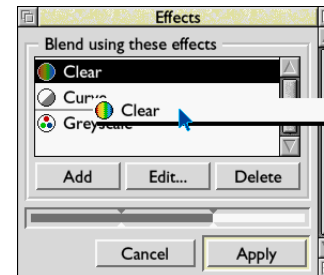
The effects are shown in a scrolling list. Effects can be selected with SELECT and de-selected with ADJUST.

The **EDIT...** and **DELETE** buttons beneath the list will highlight when the respective actions are available.

To delete the currently selected effect, click **DELETE** or press the DELETE key.

### Re-ordering Effects

The ordering of effects can be significant: drag and drop effects to re-order them. While you drag the effect a light grey indicator line will show you the position where the effect will be inserted.



## Individual Effects

### Clear

Simply clears the image to the specified colour.

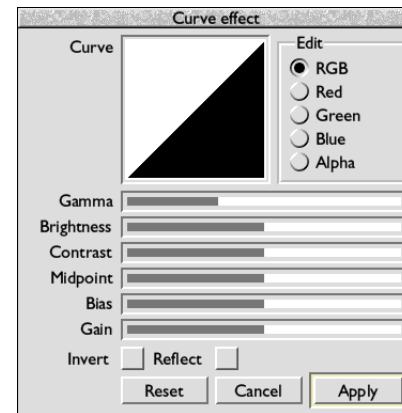
Clicking the **EDIT...** button will open up a RISC OS standard ColourPicker dialogue.

*The Clear effect is only of use by itself, as any further effects added later in the list will be applied to a plain colour.*

### Curve

The *Curve* effect offers a combined control for gamma, brightness, contrast, midpoint, bias and gain adjustment. You can also invert the curve (meaning flip left-to-right) and reflect it around its middle.

Clicking the **EDIT...** button will open up the *Curve effect dialogue*. By default it shows a linear mapping (no change):



Choose the channel to edit from the radio buttons on the top right hand side, then drag the sliders to adjust the curve.

Click **RESET** if you want the default settings back.

## Greyscale

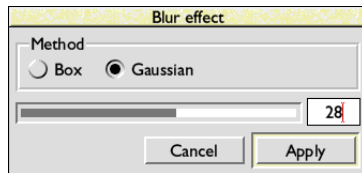
The *Greyscale* effect discards all chroma information leaving only the luma present.

As with the histogram, this uses the Rec. 601 weightings.

## Blur

The *Blur* effect offers a choice of either a box blur or a gaussian blur. Both can specify a radius of up to 47 pixels.

Clicking the **EDIT...** button will open up the *Blur effect* dialogue:



*Note that using box blur with large radii can result in visible banding.*

## Sharpen

The *Sharpen* effect provides a mild sharpening effect, using a 3x3 convolution matrix.

## Expand dynamic range

This effect looks for unused space in the shadows and highlights of the histogram and stretches the image to occupy all of the available volume.

The RGB channels are stretched by the same amount so this will preserve colours.

## Histogram equalisation

Attempts to create an even spread of values across the available colour volume.

This is useful for extracting obscured detail from images, but processes RGB separately so can lead to colour distortion.

## Examples

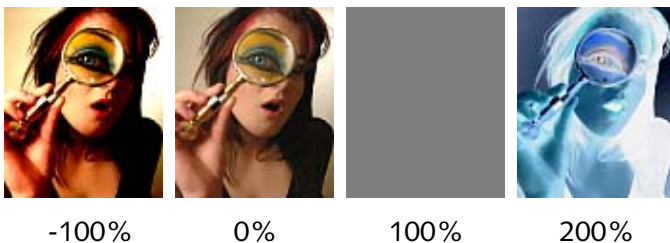
### Changing brightness

Add a Clear effect. Set it to black. Dragging the slider to the left will increase the image's brightness.



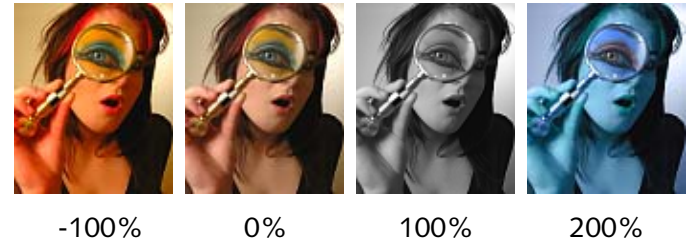
### Changing contrast

Add a Clear effect. Set its colour to 50% grey. Dragging the slider to the left will increase contrast; to the right will decrease contrast.



### Changing saturation

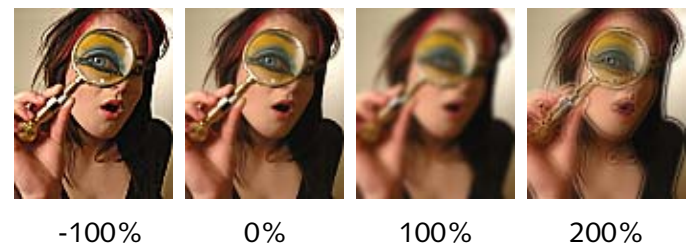
Add a Greyscale effect. Dragging the slider to the left will increase saturation; to the right will decrease saturation.



### Sharpening an image

Add a Blur effect (yes, really!) then drag the slider to the left past the left-most 'notch'.

Sharpening this way is sometimes called *unsharp mask*.





## Combined saturation and sharpening

Add a Blur effect then a Greyscale effect. Dragging the slider to the left will simultaneously increase saturation and sharpen; to the right will decrease saturation and blur.



-100%

0%

100%

200%

## Memory Use & Speed

The effects system creates two temporary bitmaps of the same size as the original image. Therefore applying effects to large images will claim a lot of memory.

Similarly large images can also take some time to process. The full image is always processed. The update after dragging the effect level slider may not always be instant.

## Further Information

The effects window's blending technique is based on "*Image Processing By Interpolation and Extrapolation*" by Paul Haeberli and Douglas Voorhies.

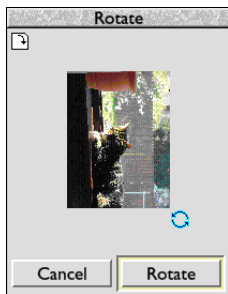
<http://www.graficaobscura.com/interp/>

# Rotate

The *Rotate dialog* can rotate Sprites and JPEGs in multiples of 90 degrees. It can also flip, transpose and transverse.

*Rotate and transform functions are only available for Sprites and JPEGs.*

Rotation and transformation of JPEGs is lossless. That means it does not perform a full decompress-transform-compress cycle which would result in quality degradation. Instead it uses *libjpeg's* lossless rotation code.



CTRL O opens the Rotate dialog. The dialog is straightforward to use. Simply drag the thumbnail image and it will spin around its centre. As you rotate the image it will snap to the nearest 90 degrees. Release when it appears in the desired orientation. Click **ROTATE** to apply

the rotation.

Dragging with ADJUST behaves similarly, except that it performs an initial horizontal flip. SELECT and ADJUST allow all possible flips and rotations to be applied.

The icon in the top left hand corner of the window shows which rotation, or transformation, will occur.

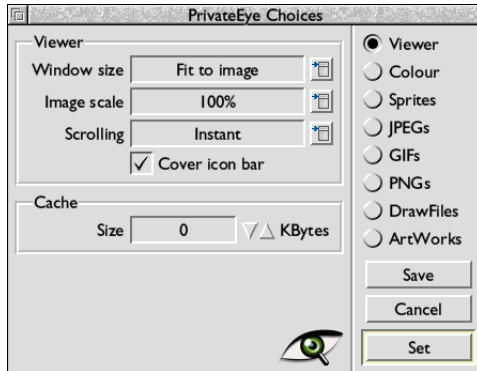
## Keys

The equivalent key presses in the viewer window are CTRL L and CTRL R to rotate left and right respectively and CTRL H and CTRL V respectively to perform horizontal and vertical flips.

# Choices

The Choices window is available from the icon bar menu. It provides control over a number of different aspects of PrivateEye's behaviour.

## Viewer Choices



**WINDOW SIZE** controls the size of the viewer window.

- **FIT TO IMAGE** sizes viewer windows to fit exactly the contained image.
- **FIT TO SCREEN** makes viewer windows at least the size of the screen.

**IMAGE SCALE** controls the image scale applied when a new

image is loaded. The choices are similar to those in the **SCALE VIEW** dialogue except for **PRESERVE** which re-uses the previously selected scale, or 100% if a new viewer window is opened.

**SCROLLING** controls how many steps are used when scrolling and zooming. The more steps you configure the smoother the effect will be. However, too many steps can also slow down if the image is complex to redraw.

PrivateEye tries to keep the viewer window in the same position when moving between images. **COVER ICON BAR** controls whether it will avoid obscuring the icon bar with viewer windows. If not set, viewer windows will be made smaller and positioned to avoid the icon bar.

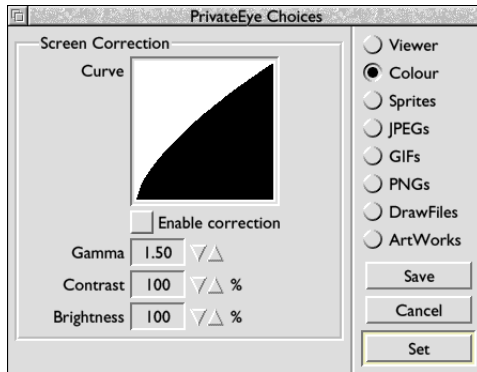
## Caching Choices

PrivateEye can set aside memory in which to retain discarded images. When you close an image and the cache is configured, then the image is moved into the cache.

**SIZE** controls the size of the cache. The default is zero which disables the cache. Choose a non-zero KByte value to enable the cache.

Use **EMPTY CACHE** on the icon bar menu to discard all cached images.

## Colour Choices



*Screen Correction* provides a simple gamma, contrast and brightness control which affects the whole desktop.

**ENABLE CORRECTION** turns on the automatic setting of these values when PrivateEye is loaded.

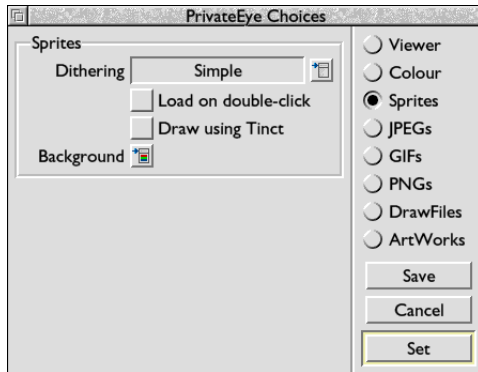
## Common File Format Choices

There are choices which exist for every file format. To avoid repetition, they are described here in advance.

**LOAD ON DOUBLE-CLICK** controls whether the respective format will be loaded when double-clicked on in a directory display.

**BACKGROUND** opens a ColourPicker dialogue which sets the background colour used for masked images and when in "fit to screen" mode. If **NONE** is chosen for the colour then a checkered background will be used.

## Sprite Choices

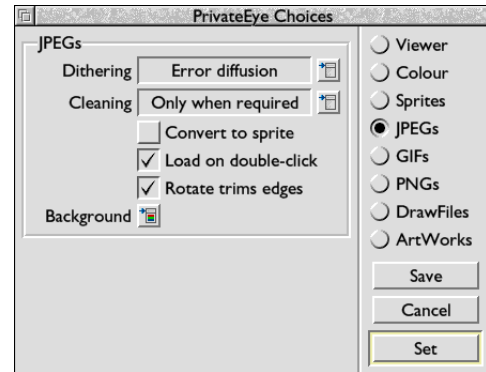


Sprite choices will affect any file format which is converted into Sprite format for display (i.e. GIFs, PNGs, JPEGs).

**DITHERING** can be set to **NONE** or **SIMPLE**. **SIMPLE** uses a stipple pattern to better approximate unavailable colours.

**DRAW USING TINCT** will draw the Sprite, where possible, using the Tinct module. Tinct offers higher quality error diffusion than the regular OS sprite plotting routines. PrivateEye uses Tinct in any case to display images with alpha channel data.

## JPEG Choices

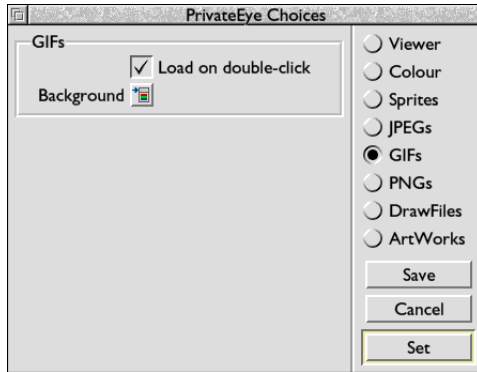


**DITHERING** can be set to **NONE**, **SIMPLE** and **ERROR DIFFUSION**.

By default JPEGs are retained in memory in their native format. If this is not desirable, perhaps to speed up redraw or to be able to use the Histogram or Editing windows, then enable **CONVERT TO SPRITE**. The JPEG will be decompressed to an 8bpp grey or 32bpp colour Sprite.

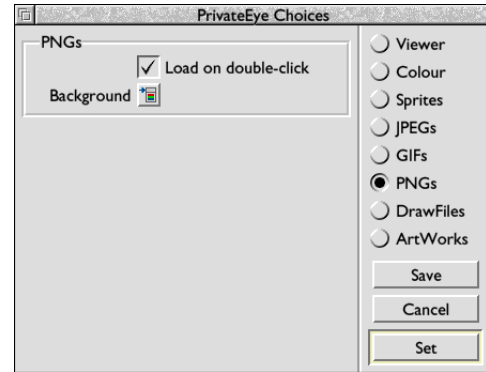
The lossless JPEG rotation code cannot transform boundary blocks. Enabling **ROTATE TRIMS EDGES** will cause those boundary blocks to be discarded in rotation.

## GIF Choices



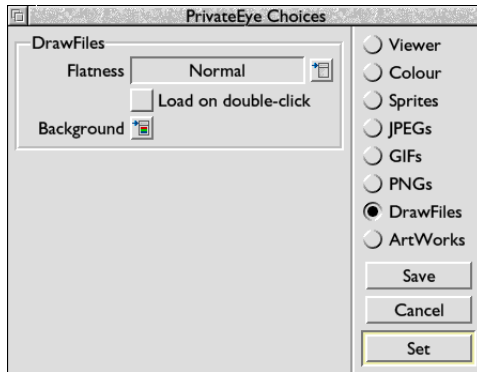
See Common File Format Choices.

## PNG Choices



See Common File Format Choices.

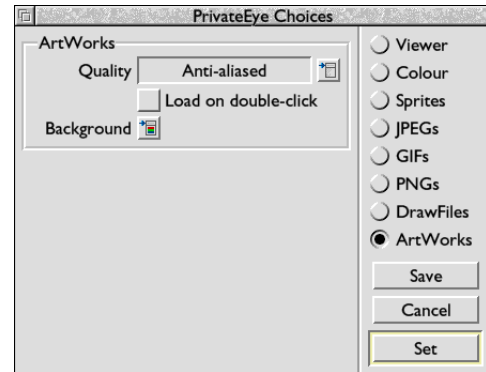
## DrawFile Choices



**FLATNESS** can be set to **COARSE**, **NORMAL**, **BEST** or **AUTOMATIC**. It controls how many lines Bézier curves are broken down into for rendering.

The DrawFile module's automatic setting does not always work well. This control is provided to allow the user to force a high-quality render when viewing at high zoom levels.

## ArtWorks Choices



**QUALITY** controls the famous ArtWorks variable quality setting. It offers **OUTLINE**, **SIMPLE**, **NORMAL** and **ANTI-ALIASED**.

# Controls

## Mouse Controls

SELECT .....ADJUST  
Gain focus.....Gain focus  
+SHIFT Zoom in to point...Zoom out from point  
+CTRL Scroll to point.....(no action)  
+DRAG Grab .....Zoom to box  
+SHIFT+CTRL Embed window.....(no action)

## Keyboard Controls

Viewer windows accept the input focus. These are the keys you can use whilst the window has the focus:

↑ ↓ ← →.....Scroll up, down, left, right  
CTRL Z.....Scroll somewhere random  
CTRL W.....Zoom in  
CTRL Q.....Zoom out  
CTRL T.....Toggle to previous zoom level  
CTRL D.....Zoom to 100%  
F11.....Open Scale dialogue  
  
CTRL L.....Rotate left  
CTRL R.....Rotate right  
CTRL H.....Horizontal flip  
CTRL V.....Vertical flip  
CTRL O.....Open Rotate dialogue

CTRL C.....Copy to clipboard

CTRL N.....New view

F3.....Open Save dialogue

CTRL S.....Convert to Sprite

CTRL I.....Open Histogram window

CTRL E.....Open Effects window

PAGE DOWN/SPACE.....Move to next file in directory

PAGE UP.....Move to previous file

CTRL F2 & ESC.....Close viewer window

ESC.....Close the viewer window

ESC.....Cancel drag (when dragging)

CTRL K.....Kill (delete file)

F1.....Start Interactive Help

SHIFT F1.....Open Image Info window

CTRL F1.....Open Source Info window

SHIFT CTRL F1.....Open Metadata window



# Keymap

PrivateEye's key assignments may be customised or supplemented by editing a file called 'Keys', which is held in PrivateEye's Choices directory:

`Choices:PrivateEye.Keys`. This usually lives inside your `!Boot` application as `!Boot.Choices.PrivateEye.Keys`.

## Sections

`Keys` is a text file built of sections. Sections begin with a square bracketed line `[Like This]` and are followed by a series of entries.

Two sections are presently used: `[Common]` and `[Viewer]`. `Common` is searched after a lookup in a more specific section is exhausted.

## Entries

Each entry within a section is of the form:

`[modifier]keyname:action`

Where `[ ]` indicates an optional part.

Tokens are case sensitive.

## Modifiers

`S_` SHIFT  
`C_` CTRL  
`SC_` SHIFT CTRL

## Special key names

`Escape`, `F1..F12`, `Print`, `Backspace`, `Tab`, `Return`, `Space`, `Logo`, `Menu`, `Insert`, `Home`, `Delete`, `Copy`, `PageUp`, `PageDown`, `Up`, `Down`, `Left`, `Right`

## Common action names

`Close`, `Help`

## Viewer action names

`ConvToSpr`, `Copy`, `Effects`, `Hist`, `HorzFlip`, `Info`, `Kill`, `NewView`, `PanDown`, `PanLeft`, `PanRandom`, `PanRight`, `PanUp`, `Rotate`, `RotateLeft`, `RotateRight`, `Save`, `Scale`, `SourceInfo`, `StepBackwards`, `StepForwards`, `VertFlip`, `ZoomIn`, `ZoomOut`, `ZoomReset`, `ZoomToggle`

## Examples

To make a TAB keypress step forwards to the next image, when in a viewer window, add the following to the [Viewer] section of a Keys file:

```
Tab:StepForwards
```

To make SHIFT TAB step backwards to the previous image:

```
S_Tab:StepBackwards
```

# History

## Version 2.99 (05 Feb 2012)

### Open sourced

This is the first release of PrivateEye since I opened up the source on riscos.info.

The source now lives at [svn://svn.riscos.info/privateeye](http://svn.riscos.info/privateeye). This is also accessible as <http://www.riscos.info/websvn/listing.php?repname=privateeye> for the web-based repository viewer.

### License updated

The old license was a bit vague which concerned the Raspberry Pi boys. I've updated it to be a BSD two-clause type license.

### Rebuilt for ARMv7 compatibility

Built using Norcroft RISC OS ARM C vsn 5.69 [20 Oct 2010] this **should** now be BeagleBoard, ARMini, etc. compatible.

### Improved blur effect

You can now choose between box or gaussian blurs and the blur radius can now be increased up to 47 pixels.

### "Midpoint" curve effect

This sets the value of the middle curve point and linearly interpolates between the extremes. It's useful for creating some tinting effects.

### FFG import improvements

Generally FFG is more reliable now.

I've fixed the spurious 'FFG translator died' message appearing when PrivateEye saw FFG messages not intended for it, e.g. when using Ovation Pro.

### Keymaps

Keymaps are now split into separate [Sections] to allow for future features. Tokens now make sense only within the appropriate section.

### libjpeg

Updated to libjpeg-8c.

### & also...

- Allowed image cache sizes up to 256M. Added PrivateEye\$DALimit to control dynamic area maximum size. See !Run.

- Added a wider range of default image scale values to the viewer choices.
- Fix: We now close the effects window when an image is entered into the cache.
- Added a little bit of anti-aliasing to the 'fx-dynamic' and 'fx-equalise' effects icons.
- PrivateEye will now build using GCCSDK.
- Numerous changes to the internals for robustness and to enable future features.

## Version 2.85 (03 Jun 2008)

### Alpha channel transparency support

Sprites and PNGs may specify an *alpha channel*. PrivateEye can now recognise and display these images. The alpha channel is assumed to contain transparency information. (Although sometimes they're used for masks rather than transparency.)

Sprites containing alpha channels in their top byte (like *Variations*, *TextEffX*, etc. use) are recognised, but RISC OS Select format alpha sprites are not yet understood.

Alpha channel images are plotted using the *Tinct* module. Tinct was previously optional but is now required by default and supplied in the release archive.

Tinct interprets alpha values of 0..255 as fully

transparent..fully opaque. Some apps may save Sprites which use the opposite sense. This area is not standardised, but you can fix it by inverting the alpha channel in the curve effects dialogue.

Some parts of PrivateEye don't fully support transparency, namely the rotation window. Transparency will be preserved, but won't be shown in the window.

### Alpha effects

The effects system has been upgraded to always preserve alpha channel data.

The curve effect has been upgraded to allow editing of the alpha channel, where present. It's also been made easier to use by turning the RGB option icons into radio icons and adding a dedicated combined RGB mode.

I've also added added a *Reflect* effect, which can be used to produce solarisation effects. The dialogue has also gained a **RESET** button.

### Other changes for alpha

The info windows now report when alpha is present.

You can now chart the alpha channel in the histogram window.

### Checkerboard pattern backgrounds

You can now choose **NONE** as the window background colour. This works for all formats. This uses Tinct to fill the window background with a checkerboard pattern.

Default choices have been changed so that bitmap backgrounds now use the checkerboard pattern.

### Improved PNG support

All of the images in PNGSuite can now be loaded correctly. <http://www.schaik.com/pngsuite/>.

1, 2, 4 and 8bpp PNGs with single transparent palette entries are now converted into masked sprites of the same depth.

bKGD chunks are used, where present, to set the viewer background colour. The image retains its transparency where present and draws on top of the background.

pHYs chunks are used, where present, to determine the correct resolution of the image. This is displayed in the source info window.

The colour/monochrome type of the image is stored and displayed in the source info window.

Errors generated by libpng whilst decoding are reported using standard Wimp error dialogues.

The unused PNG choices have been removed.

### & also...

- The Blur and Sharpen effects weren't working correctly. They were only applying the effect vertically. This now works properly and the effects are visibly stronger.
- The Colour choices 'Enable correction' option now switches off more reliably.
- The !Run file now checks for modules before attempting to RMLoad them. This gives more accurate errors when required modules are missing.
- Fixed a crash when clicking MENU on the Choices window pane workarea.

## Version 2.75 (14 Jan 2008)

### Metadata

A new *Metadata window* displays the contents of JPEG COM, Exif and Adobe segments in a tree view.

The *exiftags* library is used to decode the Exif segment.

Metadata is not available if the JPEG is converted to Sprite on loading.

### Caching

Memory can now be reserved in which to keep discarded images. This allows quick 'flips' between images when

navigating through a directory.

### **Convert to Sprite**

It wasn't very convenient to go to the Choices dialogue, enable an option and re-load a JPEG just to see its histogram or apply an effect, so you can now choose **CONVERT TO SPRITE** from the viewer menu, or use CTRL S.

### **& also...**

- Greyscale PNGs < 8bpp now load correctly.

# Copyright & License

Copyright © 1999-2012, David Thomas. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# Acknowledgements

The ArtWorks rendering module veneers are based on *ArtToSpr* by Tony Houghton.

GIF LZW decoding is based on code by Steven A. Bennett.

PNG support uses *libpng*:

Copyright © 1998-2004 Glenn Randers-Pehrson.

Copyright © 1996-1997 Andreas Dilger.

Copyright © 1995-1996 Guy Eric Schalnat, Group 42, Inc.

...and *zlib*:

Copyright © 1995-2004 Jean-loup Gailly and Mark Adler.

JPEG support uses *libjpeg*:

This software is based in part on the work of the Independent JPEG Group.

Copyright © 1991-1998, Thomas G. Lane.

Exif decoding uses *exiftags* by Eric M. Johnston.

<http://johnst.org/sw/exiftags/>

Resource location uses *ResFind* by Olaf Krumnow and Herbert zur Nedden of GAG.

Alpha sprite plotting uses *Tinct* by Richard Wilson.

<http://tinct.net/tinct.asp>

My thanks to all of the above.



# Contacting the Author

If you have any comments, bug reports or suggestions for future versions then you can contact me at [dave@davespace.co.uk](mailto:dave@davespace.co.uk).

If you have a question, then please check it has not already been answered in this documentation or the interactive help before mailing.

Please remember when reporting bugs to describe **exactly** what you were doing when the bug occurred and the version number.