

Gradgrind Manual

Gradgrind Manual • © Chris Terran 2026 • All rights reserved
Version: Gradgrind manual 101-011, Mon 16 March 2026 09:14:58 AM
Contact: www.vermilionsands.uk/roast chris@vermilionsands.uk



Gradgrind is a small RISC OS program which generates 256-entry palette files. These can either be greyscale or full colour. The program takes particular care to create very smooth blends between colours – it *grinds* out *gradients*.

Unlike most other palette utilities, you don't edit individual colours. The palette is generated by curve-fitting between control points, or *knots*. These can be interactively edited and saved as script files. A number of scripts are supplied with the program.

The greyscale palettes are particularly suitable for use as gradients in PhotoDesk. When PhotoDesk creates a palette blend, it always uses linear interpolation. This can lead to visible artefacts and banding when the gradient is applied. And it's particularly noticeable when using gradients as displacement maps.

For RGB palettes you can view in full colour, or as the individual red, green or blue channels which make it up, which lets you edit the knots. You can't change the curves in RGB mode, but other editing functions will operate on all the channels at once.

Using Gradgrind

To run the program, double-click on its icon in the Filer. A window will open where all interaction takes place. To quit the program, choose **Quit** from the menu. You can run multiple copies of Gradgrind.

Gradgrind was written and tested on RISC OS 5.31. Due to its nature, the program really should be used in a true-colour (24-bit, 16M colours) screen mode. Failing that, a 256-colour mode is usable.

Gradgrind has comprehensive interactive help.

Gradgrind uses ConfiX for its Choices. If you don't have this program the Choices file can be edited by hand.

About Interpolation

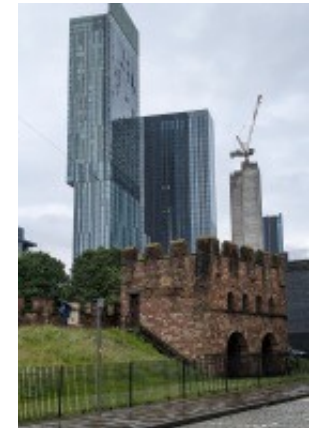
The program offers two ways of generating the colours in between the knots: *spline* and *linear*. Splines generates a smooth curve between each knot, while linear simply draws a straight line between them. PhotoDesk's palette blending is always linear.

The greyscales below show the difference. The top one uses splines – the knots are joined by curves – and the lower one uses a linear blend, as in PhotoDesk:





Using a splined palette makes a noticeable difference when using displacement maps in PhotoDesk. Here's a picture of some of Manchester's new crop of skyscrapers.

The version below on the left uses PhotoDesk's own palette blend to produce the displacement map, while the right picture uses Gradgrind's splines.



The Main Window

When Gradgrind starts up it loads a gradient file, which by default is an RGB gradient with 16 knots, all set to random values between 0.0 and 1.0. You can change the start-up file in Choices (see below).

  To create a new gradient, click on one of the top icons **1**, for RGB or greyscale. Alternatively, drop a gradient file on the window – these are text files in the standard RISC OS "Messages" format, and are fully described in the *Appendix* at the end of this document.


You can change the files used for the new gradients in Gradgrind's Choices.




You can also load and display **RISC OS palette files**, and **GIMP palettes** (text files with a .gpl extension) – see the *Loading* section below. Editing is limited for these, but you are able to export GIMP palettes as RISC OS palettes.


At the top of the window is the *palette* **2**, containing 256 different colours. Any editing will update this immediately. The palette is also displayed as *swatches* **3**, in a 16x16 grid with colour 0 at the top left. Click **Adjust** to toggle outlines of the swatches. You can save both of these as sprites by dragging them to a destination.

Channels

The large area **4** shows how each channel is defined. The icons on the left **5** allow you to display as RGB or as individual channels, and to set the interpolation method for each channel. The numbers show the number of knots in each channel.

 If RGB is selected, the colour channels are overlaid, giving an overall impression of the gradients.

   Selecting the red, green or blue icons will show that particular channel, and allow you to edit the knots. Each channel can have a different number of knots, between 4 and 30. The palette display **2** at the top of the window splits to show the entire palette on top, with the selected channel underneath.

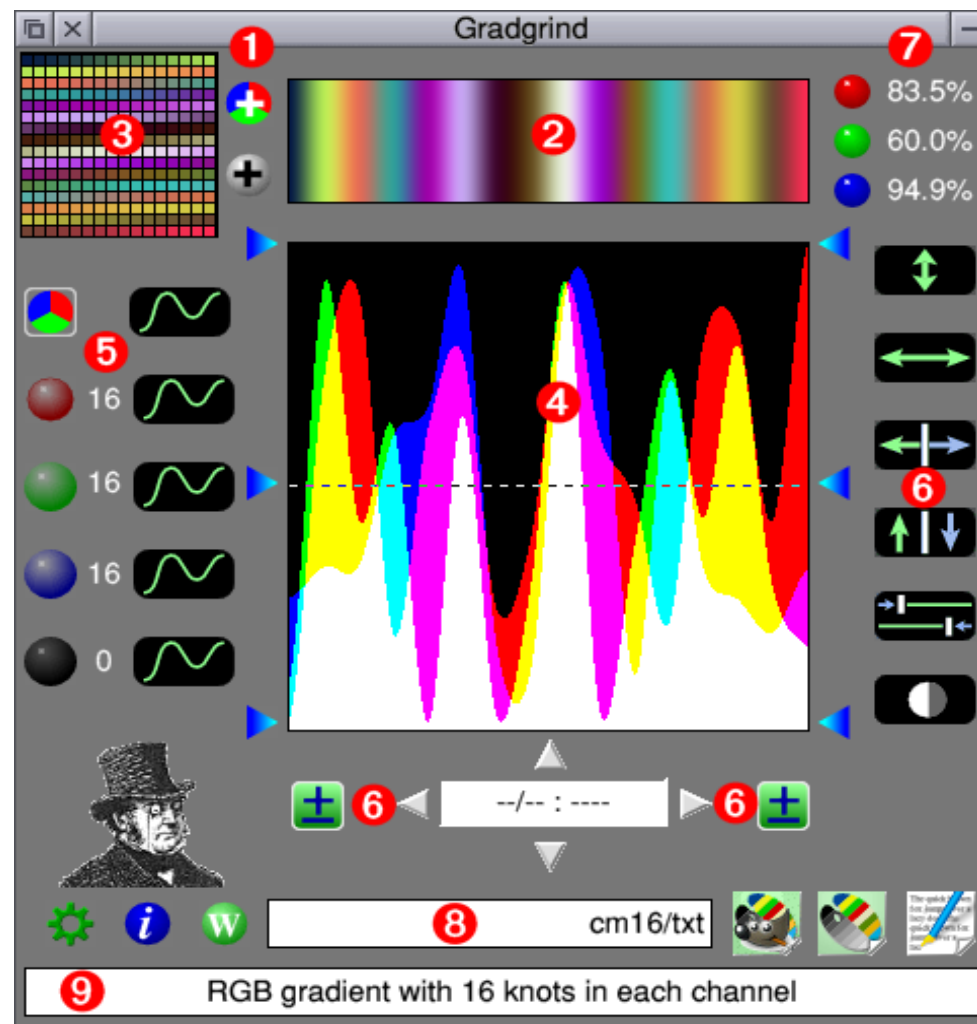
 Clicking on the greyscale icon converts the current palette to a 256-level greyscale form, using the standard weightings:

$$0.30r + 0.59g + 0.11b$$

Note this only works if each channel has the same number of knots.

The icons on the right and below **6** are for editing; see below.

When the mouse pointer is over **2** or **4**, the numbers at top right **7** show the colour values at that point. Click on any of the small coloured balls to switch between displaying them as actual pixel values in decimal or hex, percentages or real numbers. For a greyscale palette, the numbers will be the same for each channel.



The row of icons **8** below Mr Gradgrind are for **Choices**, **help** (displays this document), the [ROAST website](#), the **filename** and **saving** in various formats.

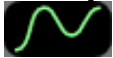
At the bottom **9** is a description of the current palette. This is writeable, and is recorded when a gradient definition file is saved.

Editing the Palette

At any time you can modify the palette globally. You can change the interpolation method for the channels, invert and mirror it in various ways, and change the overall brightness and contrast.

Any editing affects the current channel. But note that if RGB is selected all three colour channels are altered.

Interpolation



The current interpolation method is shown for each channel; they are set by the gradient definition file which was loaded. You can switch between *spline* (curve-fitting) and *linear* (straight lines). The linear method is used by PhotoDesk and most other palette editing programs.



Each channel has its own interpolation setting, so, for example, you can use a linear blend for the red channel and curve blending for blue and green.

You can change the interpolation method for any channel at any time; the result is immediately visible. Obviously this only works for gradient files, not palettes.

Mirroring



Vertical mirroring turns the palette upside down. All knots are changed to 1-old value (255-old value for palette files).



Horizontal mirroring flips the palette left-to-right. Both vertical and horizontal mirroring are reversible.



The third icon reflects the left half of the palette horizontally about its centre line. To use the right half, mirror it horizontally first. Previous values are lost.



The fourth icon makes the right half of the palette the inverse of the left – so the right half becomes the upside-down version of the left. Again, mirror first to use the other half. Previous values are lost.

Rotating



The fifth icon shifts all the knots rightward, and wraps them round so the last knot becomes the first. Press **Adjust** to shift in the other direction. If there are a different number of knots in each channel, and you're in RGB mode, each channel has its knots shifted by ± 1 , which may give unexpected colour shifts. For palette files, holding down **Shift** increases the speed.

Contrast and Brightness



The bottom icon on the right lets you modify the overall contrast and brightness of the colours. It works for palette files too.


Clicking **Select** will increase the contrast, and **Adjust** will reduce it. Use this to create a more or less pastel version of a palette.

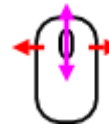
If you hold down a **Shift** key, the overall brightness is increased or decreased.

Editing Knots



Select a single-colour channel to edit the knots of a gradient file, or just display the channel for palette files. The large central image will change to display the colour channel, with the knots marked by circles.

The upper image  will split into two, with the top half showing the entire palette and the bottom half just the colour values of the selected channel.



The current knot is highlighted. To edit another knot, click **Select** or **Adjust** over the graphic, for previous/next. To change the knot's value use the **Scroll wheel**. Holding **Shift** will change faster.

Alternatively you can use the left/right bump icons to change which knot is being edited, and the up/down bump icons to change its value. **Adjust** will reverse the effect and **Shift** will increase the increment/decrement.



The central icon shows the current knot number with the total knots in the channel, followed by its value. Click on this to change the way it's displayed – it cycles between the actual knot value (0.0 to 1.0), the colour intensity in decimal (0-255) or hex (&00-&FF), and the percentage (0%-100%).

The displays will change as you modify the value.



Clicking on the plus/minus icons on either side will add or remove a knot at the start or end. **Select** adds a knot, **Adjust** deletes one. If one is added it's given a value of 0.5, which is mid-red, mid-green etc. Note that this works for RGB displays as well – all channels are affected.



The blue arrows at the sides of the knots display let you set knot values to 0.0, 0.5 or 1.0. In RGB mode, the leftmost or rightmost knots are set. When editing a channel, the currently selected knot is altered. Hold down **Shift** to set all knots.



Loading

You can load three types of file: *gradients*, *RISC OS palettes* and *GIMP palettes*. Just drop the file onto Gradgrind's window to load it.

For the palettes, editing is restricted as there are no knots present, so editing functions which affect the knots will be greyed out. You can still mirror the palette in various ways, shift it and adjust the contrast and brightness. It can be interesting to see how other palettes have been defined, by displaying each channel in turn..

Gradient files (filetype &FFF, "Text"; extension ".txt"):

These are created by Gradgrind when saving using the text icon. The extension isn't required, but will normally be added by Gradgrind so the same filename can be used for palettes, sprites and gradient.

Gradient files can be created and edited by any text editor. See the *Appendix* for details of the format.

RISC OS palettes (filetype &FED, "Palette"; extension ".pal"):

RISC OS palettes can contain up to 256 colours, and contain a sequence of VDU codes to define each colour. They can also contain other definitions such as border and mouse pointer colours, which are ignored by Gradgrind.

GIMP palettes (filetype &FFF, "Text"; extension ".gpl"):

The GIMP is a sophisticated graphics program popular on Linux (and is thus available on Raspberry Pis). Its palette files are also used by various other Linux/Windows/Mac programs (*Inkscape*, for example), and are widely available on the web; one source is the [Lospec](https://lospec.com/palette-list) site (<https://lospec.com/palette-list>). The palette save icon is still available, so you can convert these to RISC OS palettes.

Though GIMP palettes have a .gpl extension, they are just ordinary text files and can be edited as such (the file format is simple and is described [here](#)). They can contain more than 256 colours, though most don't. If there are more than 256 you can page through the palette with the left and right arrow buttons, and the description will show which colour numbers are being displayed.

GIMP colours can be named and/or commented, but this is ignored.

You should check these files in a text editor before importing them into Gradgrind. RISC OS must see the filetype as "Text", and some have very long comment lines (starting with "#") which may cause the program to fail. Gradgrind does cope with the various line endings though.

Saving

The current filename is shown next to the save icons. To save the current gradient, edit the filename appropriately and drag one of the icons at the right to a filer window, or directly to another application's window.

By default, the filename will have an extension added – */pal*, */gpl*, */txt* or */spr* – so that you can use the same filename for each type of save. You can change this behaviour in Gradgrind's Choices. If you do, take care not to overwrite files.



Drag the *text* icon to save as a Gradgrind script file. This means you can always regenerate the palette and edit it again. See the *Appendix* for details of the format used. Any text in the bottom-most icon is included in the file as the **"description:"** tag's value. If you've loaded a palette file this will be greyed out.



Drag the *palette* icon to save as a RISC OS palette file. If you have PhotoDesk running you can drop the icon directly onto PhotoDesk's own palette window, or on the gradient dialogue.



Drag the *GIMP palette* icon to save as a GIMP palette file. These are plain text files and can be edited as such. They are given a */gpl* extension and a "Text" filetype. If you've loaded a large GIMP palette files (some have over 3,000 colours), note that only the currently shown page is saved, 256 colours.



Dragging the *gradient* or the *swatches* displays will save them as a sprite. A "G" or "S" is appended to the name, as appropriate. It's useful as a visual reminder of the gradient.

Choices

Click on the cogwheel to edit Gradgrind's Choices. The program uses ConfiX for this. Change any values and click on **Save**, and they will be immediately visible.

If you don't have ConfiX a text file is shown which you can edit and save. In this case you will need to restart Gradgrind for any changes to take effect. You can get ConfiX from <http://www.xat.nl/riscos/sw/confix/>

You can set the start-up file, and which files are used for the New RGB and New Greyscale icons. Adding a file extension is the default (and is recommended), but you can change this if you wish.

Default display options for various numbers can be set, as can the colours used for each channel and whether swatches are outlined.

Technical Notes

- In spline mode, the colours are interpolated using a Catmull-Rom spline function. This has the property that the curve passes through each point. See https://en.wikipedia.org/wiki/Centripetal_Catmull%E2%80%93Rom_spline for more information.
- If any errors occur when reading gradient scripts, the palette will show red, green and blue vertical stripes.
- The start-up and new files available in Choices use a naming convention. The first letter is C for colour or G for greyscale, followed by the number of knots – a single number if they're all the same, or three 2-digit numbers for r, g and b if they're different. Any description follows. So *G12wave.txt* is a greyscale gradient with 12 knots, and *C041608pastel.txt* is an rgb gradient with 4 red knots, 16 green and 8 blue.
- The program's logo is, of course, Mr Gradgrind from Dickens' *Hard Times*. It was drawn by Harry French for the first edition of the book in 1854.

Acknowledgements

Thanks to:

All at [R-Comp](#), especially Andrew Rawnsley

All at [RISC OS Open](#), and its forum contributors

[X-Ample](#) for ConfiX and work on PhotoDesk

Appendix: Gradient Script File Format

Gradient definitions are stored as normal text files. They use the standard "messages" file format of tag:value pairs on separate lines. Load one by dropping it on Gradgrind's main window.

Here is a sample gradient file:

```
# Lines starting with a # are comments
# StrongED$Mode = Message
# The line above tells StrongED how to display this file
gradient_version:1.00
format:rgb
knot_countr:12
knotsr:RND(1)
knot_countg:4
knotsg:0.25
knot_countb:8
knotsb:RND(1)/2+0.25
```

Lines starting with # are ignored.

Tags can be in any order.

The gradient_version tag is required. The file is recognised as a valid gradient file by this tag. It should have a value of (exactly) 1.00.

The format tag is also required, and must be either rgb (for a full-colour palette) or greyscale (for a monochrome greyscale palette).

The remaining tags define the palette by specifying the number of knots in each channel, and their values. Here, knot_countr says that there are 12 knots in the red channel. Knots are numbered from 1 to knot_count.

Knot values must be real numbers between 0.0 and 1.0, with 0.0 representing zero intensity (black) and 1.0 maximum. You could set the red knots individually with knotr1:val, knotr2:val and so on, but here we're using the knotsr tag, which sets all red knots, in this case to a random number between 0 and 1.0.

All values for knots are *evaluated*, so you can use any valid BASIC expression. In the evaluation you can reference the variable k%, which holds the current knot number being processed.

The other channels – and they all must be specified – are defined similarly, with knot_countg and knot_countb defining the number of knots in the green and blue channels, and knotsg and knotsb giving them values. Every knot in the green channel has the same value of 0.25, and all the blue knots have a random number between 0.0 and 0.5 (the division by 2) added to 0.25 – so it ends up between 0.25 and 0.75.

Gradient file tags

Each tag is specified below; they can be in any order.

#

Lines starting with this character are ignored, so can be used to comment the file.

gradient_version:1.00

Required. and must have a value of 1.00.

Example:

```
gradient_version:1.00
```

format:rgb | greyscale

Required. This must be either rgb or greyscale.

Examples:

```
format:rgb
format:greyscale
```

Other values cause an error.

description:text

Optional. A short description of the gradient. When you save a gradient, the contents of the writeable icon at the bottom of the display is stored as the value of this tag. If the tag doesn't exist, or you've loaded a RISC OS or GIMP palette, some general information is shown.

knot_count[suffix]:expr

Required. This specifies the number of knots in a channel, and can have various suffixes. The *expr* must result in an integer between 4 and 30. It is evaluated, so can access user (*n*) variables.

On its own, with no suffixes, this is used for greyscale palettes.

Example:

```
format:greyscale
knot_count:8
```

For an RGB palette, there must be a setting for each channel, marked by an r, g or b suffix.

Example:

```
format:rgb
knot_countr:4
knot_countg:8
knot_countb:12
```

interpolation[suffix]:spline | linear

Optional. This specifies the interpolation type of each channel, and can be either spline (the default, if unspecified) or linear.

With no suffix the greyscale interpolation is set, otherwise the suffix should be r, g or b.

Example:

```
format:rgb
# The green channel uses linear interpolation, the
# others default to spline
knot_countr:4
knot_countg:8
interpolationg:linear
knot_countb:12
```

knot[suffix]:expr

Optional. These tags define each knot. The *expr* must result in a real number between 0.0 and 1.0, with 0.0 representing zero intensity and 1.0 the maximum. If it is outside this range it is forced to be between 0.0 and 1.0. If a knot isn't defined it's given a value of 0.5.

The *expr* is *evaluated*, and can be any BASIC expression. While the evaluation is taking place, the variable *k%* holds the current knot number, from 1 to knot_count. You can also use the user (*n*) values (see below). You should take care with the expression, as it's evaluated.

For greyscale palettes, the suffix is simply the knot number, or use knots to specify every knot. The knots tag takes precedence; if it's present, any individual knot definitions are ignored.

Examples:

```
format:greyscale
knot_count:4
# Specify each greyscale knot individually
knot1:0.4
knot2:RND(1)/2
knot3:0.22
knot4:1.0
```

This uses the knots form to set all the knots:

```
format:greyscale
knot_count:5
#Set all 5 knots to a random value
knots:RND(1)
```

This example only supplies a couple of knot values, with the others set to 0.5:

```
format:greyscale
knot_count:16
# Any missing knot values default to 0.5
knot1:0.1
knot2:0.3
knot15:0.7
knot16:0.9
```

As knot values default to 0.5 if not supplied, the shortest possible gradient file is thus:

```
gradient_version:1.00
format:greyscale
knot_count:8
```

which produces a rather boring mid-grey palette, with every knot – and hence every shade of grey in the palette – the same.

For an rgb palette you specify the knots for each channel. To specify them individually, the suffix should be the channel (r, g or b) followed by the knot number (1 to knot_countchannel inclusive). So knotg2:0.231 sets the second knot of the green channel to 0.231.

There is a special form of this tag, which lets you specify a single value for every knot in a channel: knotschannel. Note the plural! So knotsb:0.8 would set all knots in the blue channel to 0.8, which is a medium-strong blue.

Example:

```
format:rgb
knot_countr:8
knot_countg:4
knot_countb:12
# Set all 8 red knots to 0.25
knotsr:0.25
# Specify each green knot individually
knotg1:0.231
knotg2:1.0
knotg3:0.839
knotg4:0.5
# All blue knots are random
knotsb:RND(0)
```

For obvious reasons, the knots forms will never be produced when saving a Gradgrind gradient file; the knot values will all be specified individually.

user[suffix]:expr

Optional. These tags let you define some real constants to use in the gradient definition. There are 10 user variables, defined as user0:value to user9:value. They are stored in a real array as user(0) to user(9) and can be referenced as such when defining knot values. The value must be a real number between 0.0 and 1.0, and is evaluated just once, when the user value is defined.

Examples:

```
# A pastel-toned palette – adjust the user variables to suit
format:rgb
# user(0) is scale factor
user0:2.3
# user(1) is offset
user1:0.4
# Number of knots...
knot_countr:12
knot_countg:12
knot_countb:12
# ...and their values
knotsr:RND(1)/user(0)+user(1)
knotsg:RND(1)/user(0)+user(1)
knotsb:RND(1)/user(0)+user(1)
```

This one generates a simple waveform. Knot values alternate between user(0) and user(1), depending on whether k% – the knot being processed – is even or odd. The number of knots is set in user(9). You could cycle between three or more values by altering the MOD specification.

```
format:greyscale
# We alternate between user(0) and user(1)
user0:0.6
user1:0.4
# Number of knots...
knot_count:16
# ...and their values
knots:user(k% MOD 2)
```