

**NAME**

`gdk-pixbuf-csource` - C code generation utility for GdkPixbuf images

**SYNOPSIS**

`gdk-pixbuf-csource` [OPTIONS...] <IMAGE>

**DESCRIPTION**

`gdk-pixbuf-csource` is a small utility that generates C code containing images, useful for compiling images directly into programs.

`gdk-pixbuf-csource` either takes as input one image file name to generate code for, or, using the `--build-list` option, a list of (**name**, **image**) pairs to generate code for a list of images into named variables.

This tool is mostly meant to be used for backward compatibility. Newly written applications and libraries should use GResource to embed image assets in their binary.

**OPTIONS****--stream**

Generate pixbuf data stream: a single string containing a serialized **GdkPixdata** structure in network byte order.

**--struct**

Generate **GdkPixdata** structure; your code needs the **GdkPixdata** structure definition from **gdk-pixdata.h**.

**--macros**

Generate **\*\_ROWSTRIDE**, **\*\_WIDTH**, **\*\_HEIGHT**, **\*\_BYTES\_PER\_PIXEL** and **\*\_RLE\_PIXEL\_DATA** or **\*\_PIXEL\_DATA** macro definitions for the image.

**--rle**

Enables run-length encoding for the generated pixel data (default).

**--raw**

Disables run-length encoding for the generated pixel data.

**--extern**

Generate extern symbols.

**--static**

Generate static symbols (default).

**--decoder**

Provide a `*_RUN_LENGTH_DECODE(image_buf, rle_data, size, bpp)` macro definition to decode run-length encoded image data.

**--name=identifier**

Specifies the identifier name (prefix) for the generated variables or macros (useful only if **--build-list** was not specified).

**--build-list**

Enables (**name**, **image**) pair parsing mode.

**-h, --help**

Prints a brief help and exit.

**-v, --version**

Prints the tool version and exit.

**--g-fatal-warnings**

Makes warnings fatal, and causes the program to abort.

**SEE ALSO**

The **GdkPixbuf** documentation, shipped by gdk-pixbuf, and also available online on *docs.gtk.org*.

**BUGS**

The runlength encoder gets out of sync with the pixel boundaries, since it includes the rowstride padding in the encoded stream. Furthermore, it generates pixbufs with suboptimal rowstride in some cases.