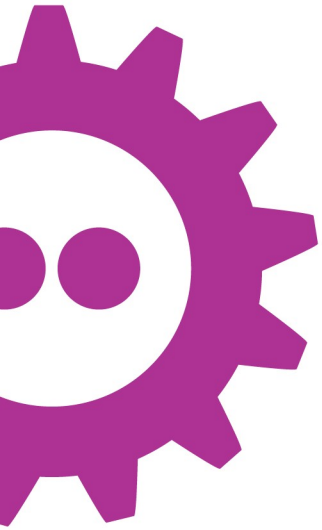


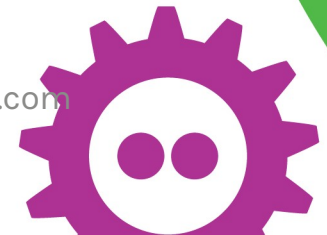
A Pixel Format Guide

to the galaxy

Alexandros Frantzis

alexandros.frantzis@collabora.com

3/2/2018

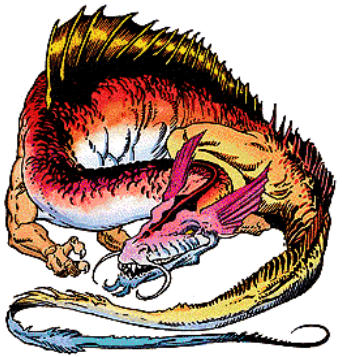


FOSDEM 18

COLLABORA

Introduction

GL_BGRA+GL_UNSIGNED_INT_8_8_8_8



DRM_FORMAT_RGB565

SDL_PIXELFORMAT_RGB332

QImage::Format_RGB30

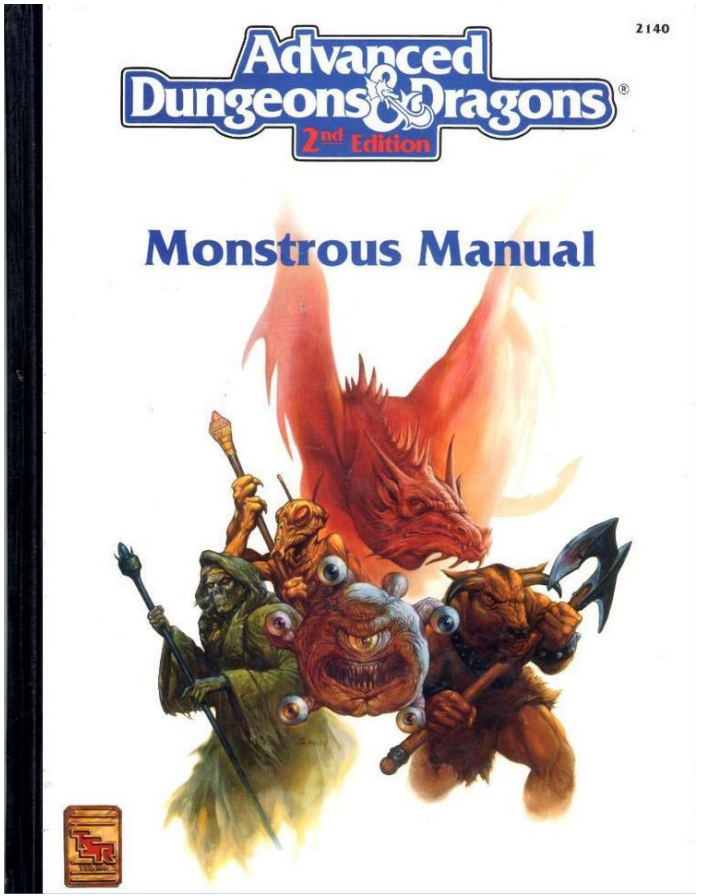
DSPF_RGB32

VK_FORMAT_A2R10G10B10_UNORM_PACK32

COLLABORA

Don’t panic!

- Component layout
- Packed vs Array formats



COLLABORA

Pfg documentation

- High-level descriptions of pixel format families
- Github website

DRM pixel formats

The drm pixel formats follow the scheme:

```
DRM_FORMAT_{component-format}
```

The `component-format` part specifies the order and sizes of the components. All the components are listed first, optionally followed by their corresponding sizes in bits (e.g., `RGB888`, `YUVY`).

There are a few categories of drm formats:

- RGBA formats: `DRM_FORMAT_{rgba-componenent-format}`

  The `rgba-component-format` specifies the order and sizes of the components in a native type **on a little-endian system**, with the leftmost component stored in the most significant bits, and the rightmost in the least significant bits.

  Although the formats are expressed in terms of a native type, the addition of little-endianness specification makes the format endianness-independent.

  For formats where components occupy whole bytes, an alternative, and probably more straightforward way to express the expected order, is that the `component-list` specifies the order of the components in memory with the leftmost component at the highest address and the rightmost at the lowest addres.

  Example: `DRM_FORMAT_ARGB8888`

  Always stored as B, G, R, A in memory (B at lowest address, A at highest)

  Example: `DRM_FORMAT_RGB565`

  Always stored in memory as:

```
0               1
M             L M             L
G2G1G0B4B3B2B1B0 R4R3R2R1R0G5G4G3
```

Pfg tool intro

```
$ python3 -m pfg describe [OPTIONS...] [FORMAT]
$ python3 -m pfg find-compatible [OPTIONS...]
                                 [FORMAT] [FAMILY]
$ python3 -m pfg document [FAMILY]
$ python3 -m pfg list-families
$ python3 -m pfg list-formats [FAMILY]
```

COLLABORA

Pfg tool: describe

```
$ python3 -m pfg describe PIXMAN_r5g6b5
Format:                  PIXMAN_r5g6b5
Component data type:     UNORM
Described as:            Native 16-bit type
Native type:             M                                 L
                         R4R3R2R1R0G5G4G3G2G1G0B4B3B2B1B0
Memory little-endian: 0                  1
                         M                L M                 L
                         G2G1G0B4B3B2B1B0 R4R3R2R1R0G5G4G3
Memory big-endian:    0                  1
                         M                L M                 L
                         R4R3R2R1R0G5G4G3 G2G1G0B4B3B2B1B0
```

COLLABORA

Pfg tool: describe (2)

```
$ python3 -m pfg describe DRM_FORMAT_RGB565
Format:                 DRM_FORMAT_RGB565
Component data type:    UNORM
Described as:           Bytes in memory
Memory little-endian: 0                   1
                      M                 L M                   L
                      G2G1G0B4B3B2B1B0 R4R3R2R1R0G5G4G3
Memory big-endian:    0                   1
                      M                 L M                   L
                      G2G1G0B4B3B2B1B0 R4R3R2R1R0G5G4G3
```

COLLABORA

Pfg tool: find-compatible

```
$ python3 -m pfg find-compatible
    VK_FORMAT_R8G8B8A8_UNORM sdl2
Format: VK_FORMAT_R8G8B8A8_UNORM
Is compatible on all systems with:
        SDL_PIXELFORMAT_RGBA32
Is compatible on little-endian systems with:
        SDL_PIXELFORMAT_ABGR8888
Is compatible on big-endian systems with:
        SDL_PIXELFORMAT_RGBA8888
```

COLLABORA

Pfg tool: find-compatible (2)

- Flags for more relaxed matching
  - --treat-x-as-a
  - --treat-srgb-as-unorm
  - --ignore-data-types

COLLABORA

Pfg as a library

```
import pfg
desc = pfg.describe(format_str)
comp = pfg.find_compatible(format_str, family_str,
                           treat_x_as_a=False,
                           treat_srgb_as_unorm=False,
                           ignore_data_types=False)
doc = pfg.document(family_str)
families = pfg.list_families()
formats = pfg.list_formats()
```

COLLABORA

Pfg as a library (2)

```
import pfg
print("vk_format = VK_FORMAT_UNDEFINED;")
print("switch(sdl2_format) {")
for f in pfg.list_formats("sdl2"):
    comp = pfg.find_compatible(f, "vulkan")
    match = len(comp.everywhere) > 0 or len(comp.little_endian) > 0 or \
            len(comp.big_endian) > 0
    if match: print("case %s:" % f)
    if len(comp.everywhere) > 0:
        print("  vk_format = %s;" % comp.everywhere[0])
    else:
        if len(comp.little_endian) > 0:
            print("  if (little_endian) vk_format = %s;" % comp.little_endian[0])
        if len(comp.big_endian) > 0:
            print("  if (!little_endian) vk_format = %s;" % comp.big_endian[0])
    if match: print("  break;")
print("}")
```

COLLABORA

Pfg as a library (3)

```
vk_format = VK_FORMAT_UNDEFINED;
switch(sdl2_format) {
case SDL_PIXELFORMAT_RGBA4444:
  vk_format = VK_FORMAT_R4G4B4A4_UNORM_PACK16;
  break;
case SDL_PIXELFORMAT_RGB888:
  if (little_endian) vk_format = VK_FORMAT_B8G8R8_UNORM;
  if (!little_endian) vk_format = VK_FORMAT_R8G8B8_UNORM;
  break;
case SDL_PIXELFORMAT_ARGB8888:
  if (little_endian) vk_format = VK_FORMAT_B8G8R8A8_UNORM;
  break;
case SDL_PIXELFORMAT_RGBA8888:
  if (!little_endian) vk_format = VK_FORMAT_R8G8B8A8_UNORM;
  break;

                                        ...

}
```

Adding a pixel format family

- Documentation in docs/***family***.md
- Test-driven approach, tests in tests/test_***family***.py
- Code in pfg/***family***.py
  - describe(format_str) → FormatDescription
  - formats() → list of supported format strings
  - document() → Returns documentation from docs/***family***.md
- Utility functions in pfg/util.py
- More details in README.md

COLLABORA

Current state and future

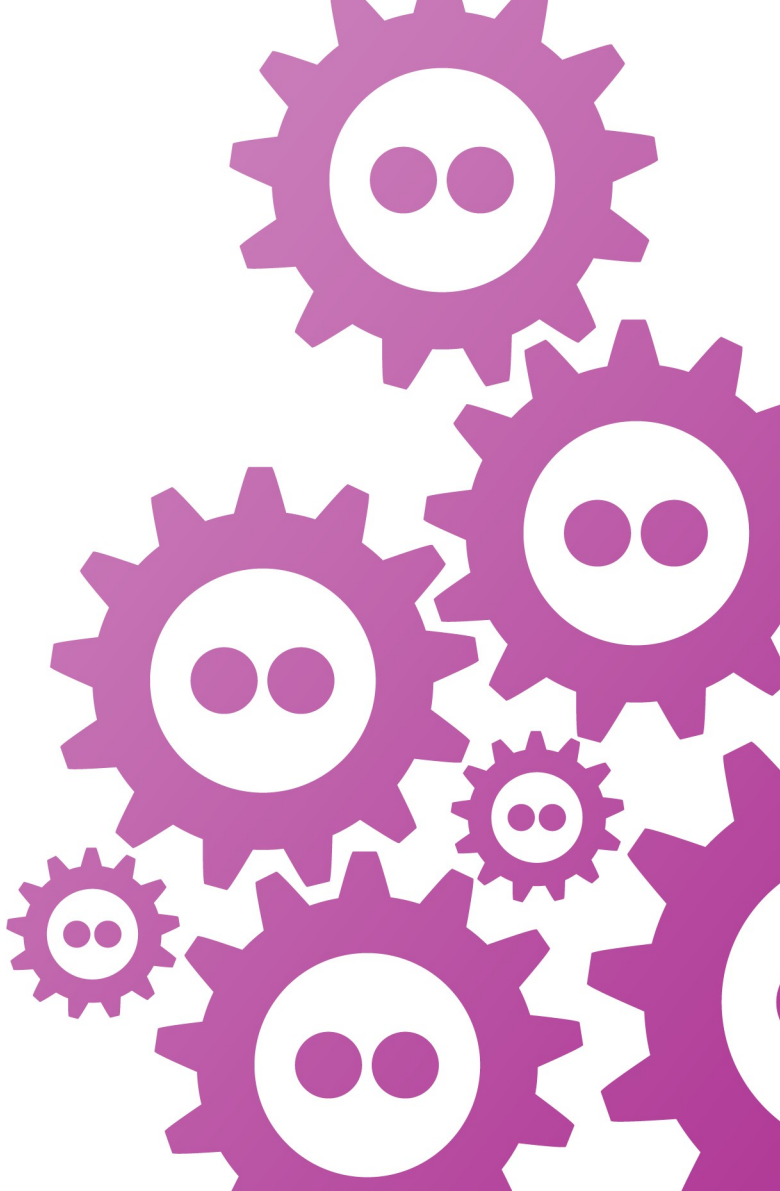
- 12 pixel format families, over 450 pixel format names
- More pixel format families!
- Compressed and multi-plane formats (for matching).

COLLABORA

FOSDEM 18

A Pixel Format Guide to the galaxy

Any questions?



COLLABORA

Links

- Project page: https://afrantzis.github.io/pixel-format-guide/
- Project repo: https://github.com/afrantzis/pixel-format-guide
- Blog: https://afrantzis.wordpress.com