Embedded Graphics Drivers in Mesa

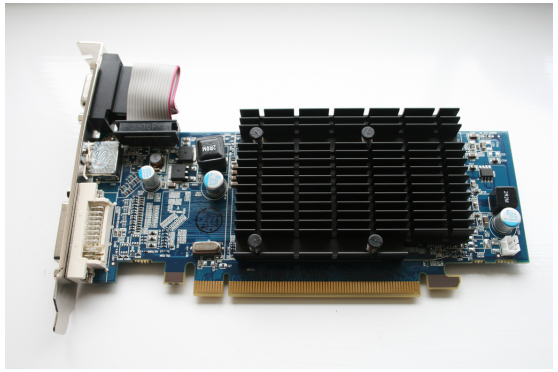
Neil Roberts

igalia

Overview

About GPUs

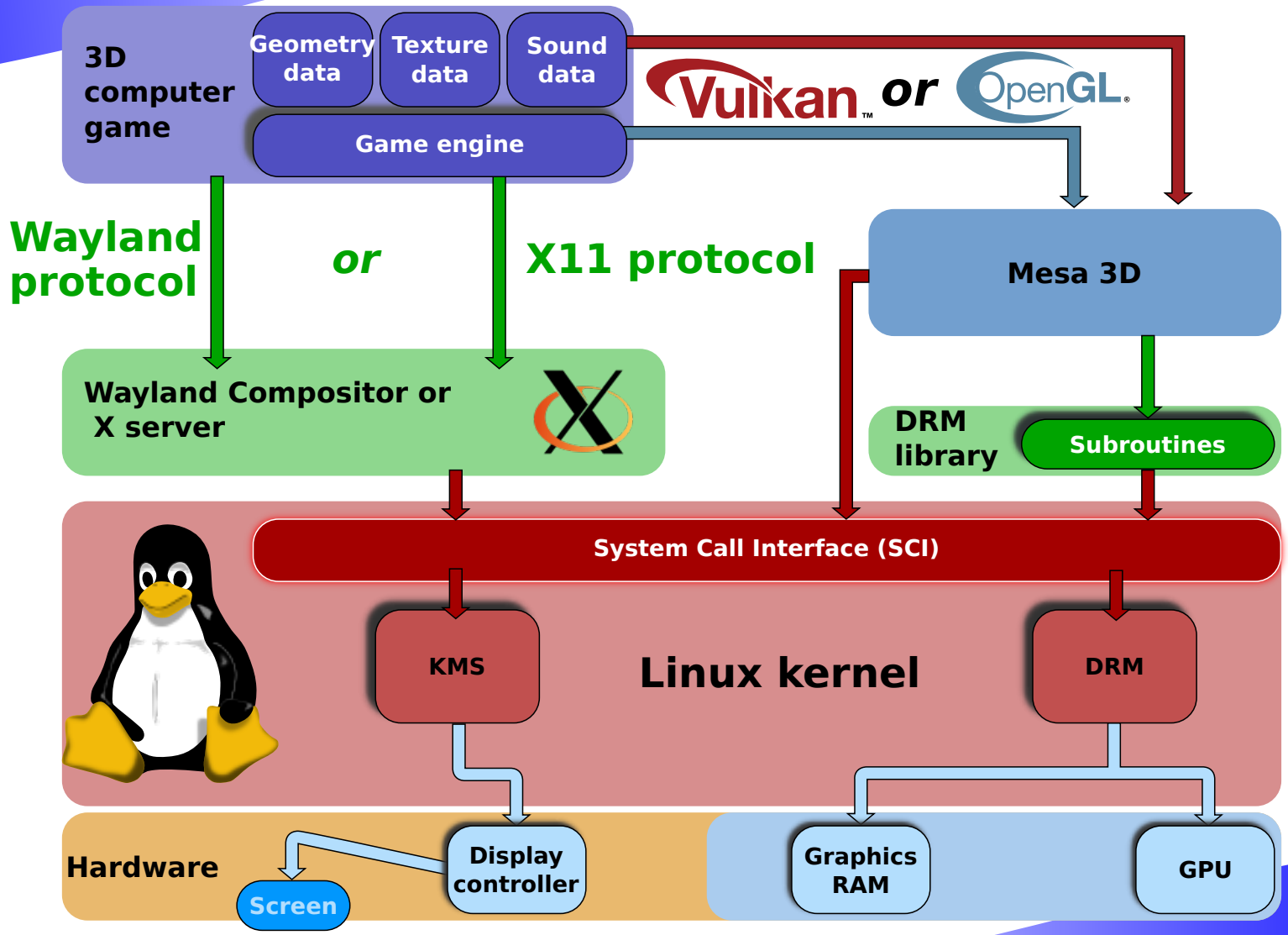
- It is a specialized electronic circuit designed to rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display device. Wikipedia.



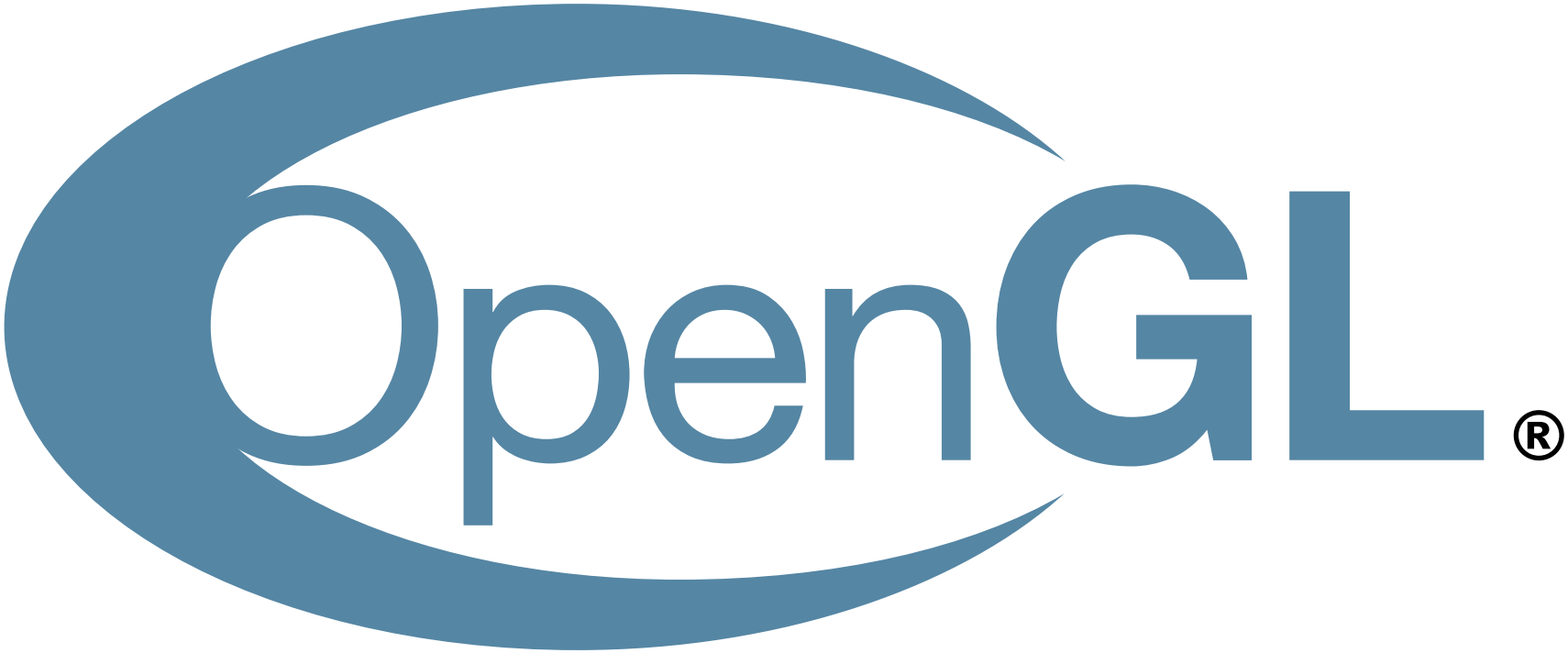
- They are becoming increasingly general purpose processors that can run programs (shaders).
- They are highly threaded and typically use SIMD to operate on multiple inputs at the same time.
- Still contain fixed function pieces for graphics-specific functions:
  - Texture sampling
  - Primitive assembly
  - etc

Linux graphics stack

based on diagram by Shmuel Csaba Otto Traian; GNU FDL 1.3+ and CC-BY-SA 3.0+; created 2014-03-02, last updated 2015-05-30



Graphics APIs


OpenGL®

- OpenGL 1.0 was released in January 1992 by Silicon Graphics (SGI).
- Based around SGI hardware of the time which had very fixed functionality.
- Eg, explicit API to draw a triangle with a colour:

```
/* Set a blue colour */
glColor3f(0.0f, 0.0f, 1.0f);
/* Draw a triangle, describing its points */
glBegin(GL_TRIANGLES);
 glVertex3f(0.0f,1.0f,0.0f);
 glVertex3f(-1.0f,-1.0f,0.0f);
 glVertex3f(1.0f,-1.0f,0.0f);
glEnd();
```

- In 2004 OpenGL 2.0 was released.
- Introduced the concept of shaders.
- Can now influence the rendering with programs called shaders.
- Eg, choose a colour programatically:

```
void main()
{
        /* Choose the colour based on the X-position of the pixel */
        gl_FragColor = vec4(gl_FragCoord.x * 0.008 - 1.0, 0.0, 0.0, 1.0);
}
```

- In later versions of GL more and more functionality is moved into the programmable shaders.
- Much more programmable, much less fixed-function.
- Inputs are more often given in buffers rather than via API calls.
- Eg, vertex data now in a buffer:

Buffer containing vertices

```
# Position      Colour
-1 -1           0xff0000ff
0  -1           0xff0000ff
-1 0            0xff0000ff
0  -1           0xff0000ff
-1 0            0xff0000ff
0  0            0xff0000ff
```

Commands describing buffer layout

```
glVertexAttribPointer(0, 2, GL_FLOAT,
                      GL_FALSE, 12, 0);
glVertexAttribPointer(1, 4, GL_UNSIGNED_BYTE,
                      GL_TRUE, 12, 8);
```

OpenGL ES

- Simplified version of OpenGL targetting embedded devices.
- Removes most of the legacy cruft and things that are hard to implement in hardware.
- Is increasingly similar to modern versions of OpenGL which also try to deprecate old functionality.


Vulkan
TM

- Vulkan 1.0 released in 2016
- Clean break from legacy OpenGL
- Much less driver overhead
- Everything is specified in buffers
- The application has the responsibility to manage buffers and synchronisation.
- Harder to use but allows applications to exploit the hardware better

History of Mesa

Architecture of Mesa

Embedded drivers

Freedreno

Panfrost

Broadcom

Thanks.

Questions?