Dzmitry Malyshau

07/7/2017

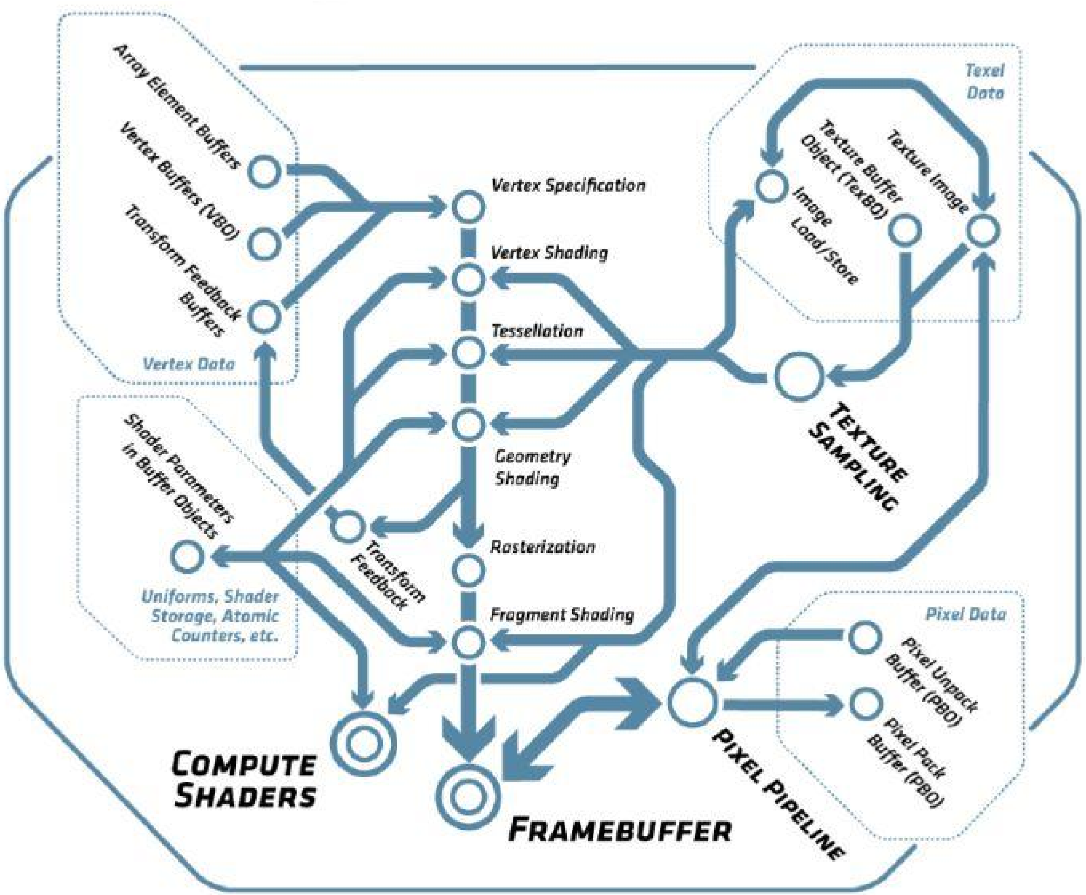
Rust Toronto

# Graphics…

…

…

# no time to explain!



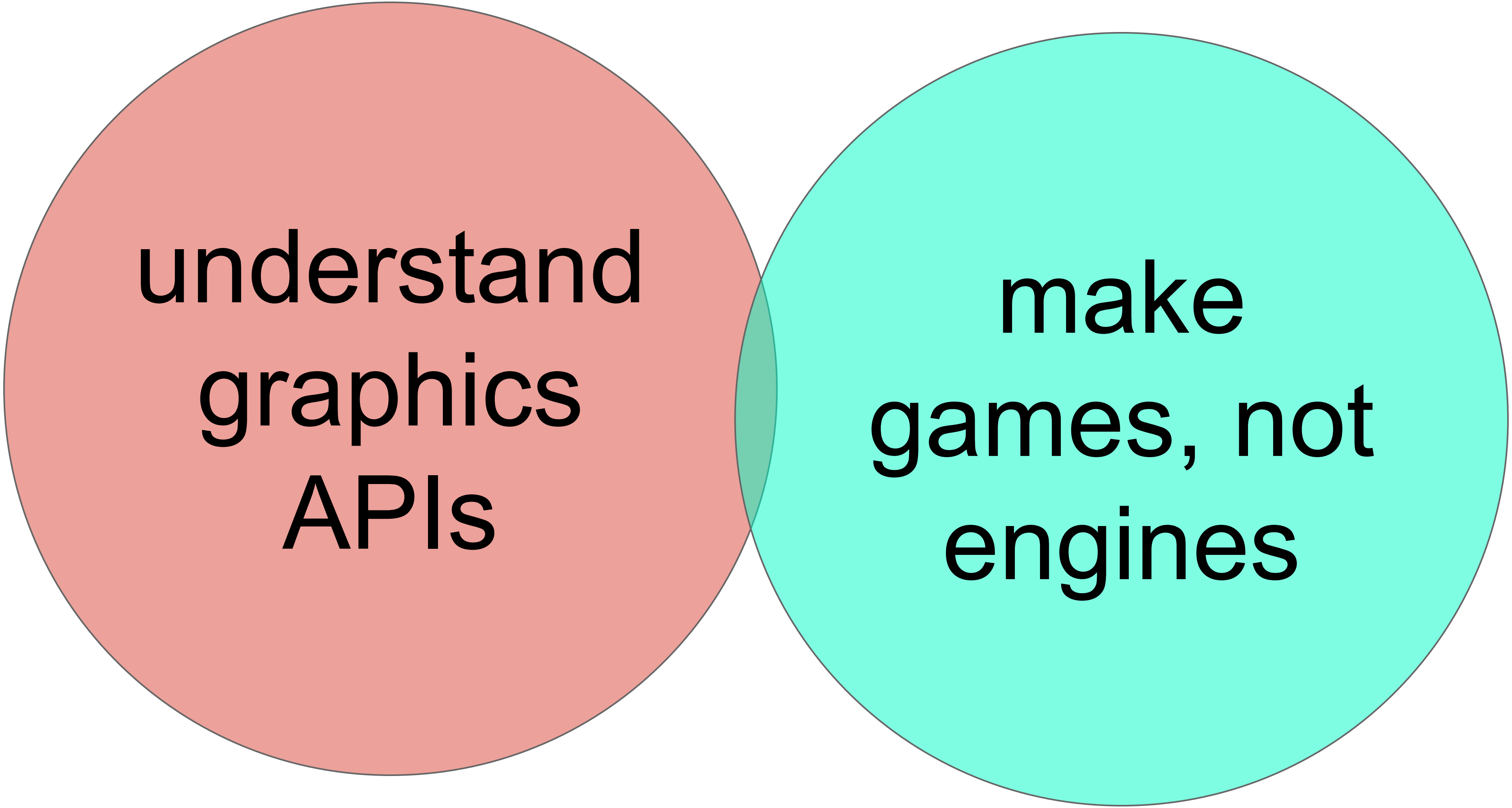
# Existing engines?

- Piston
- Amethyst
- GGEZ
- Kiss3D
- … graveyard of abandoned stuff

# Rendering with gfx-rs

- Examples
- Tutorials

… the rest
of the game

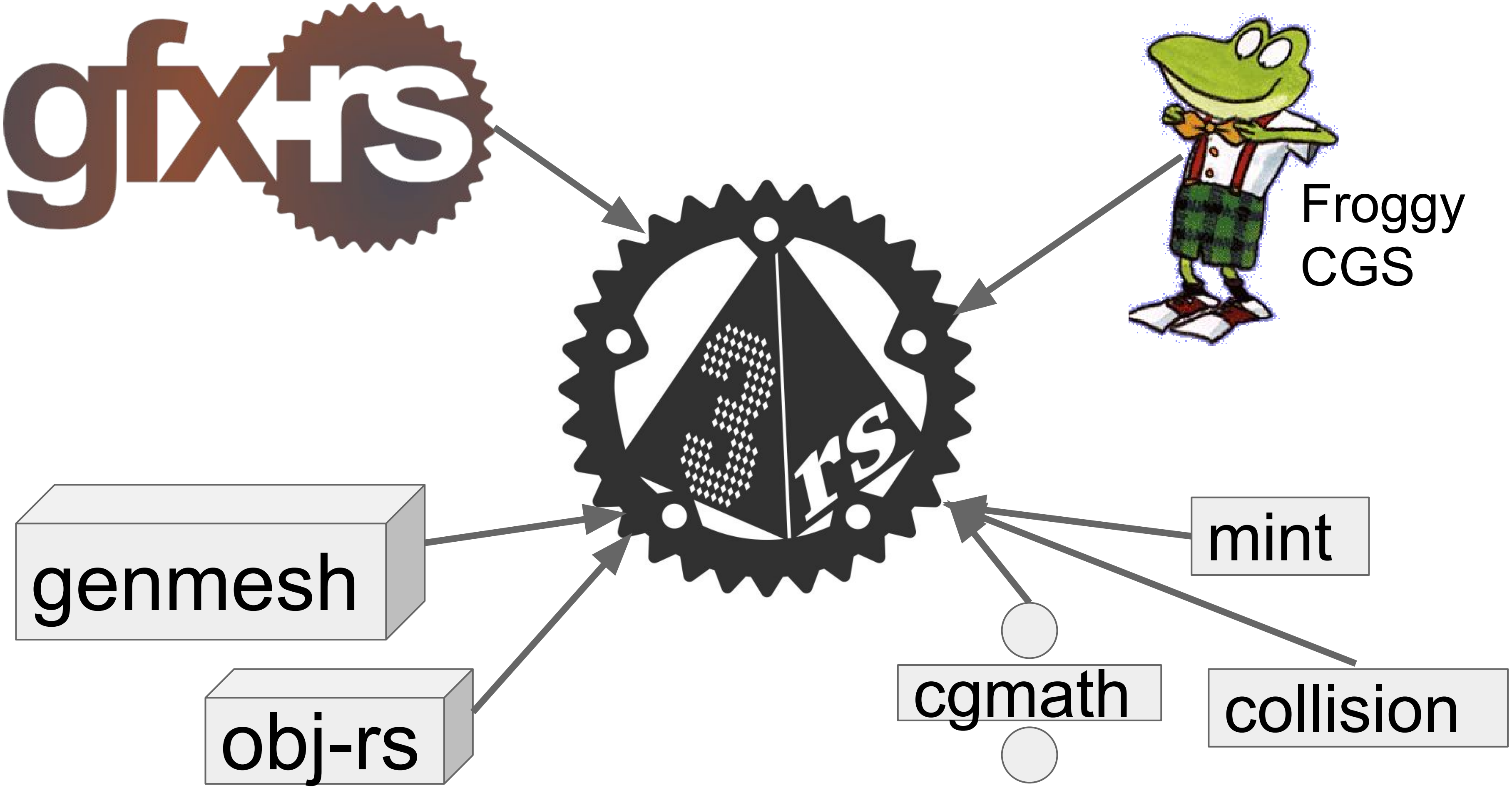


understand graphics APIs

make games, not engines

# Three...JS ???

- (almost) the standard of Web graphics
- ultimate prototyping weapon
- not because it’s easy...



Froggy
CGS

genmesh

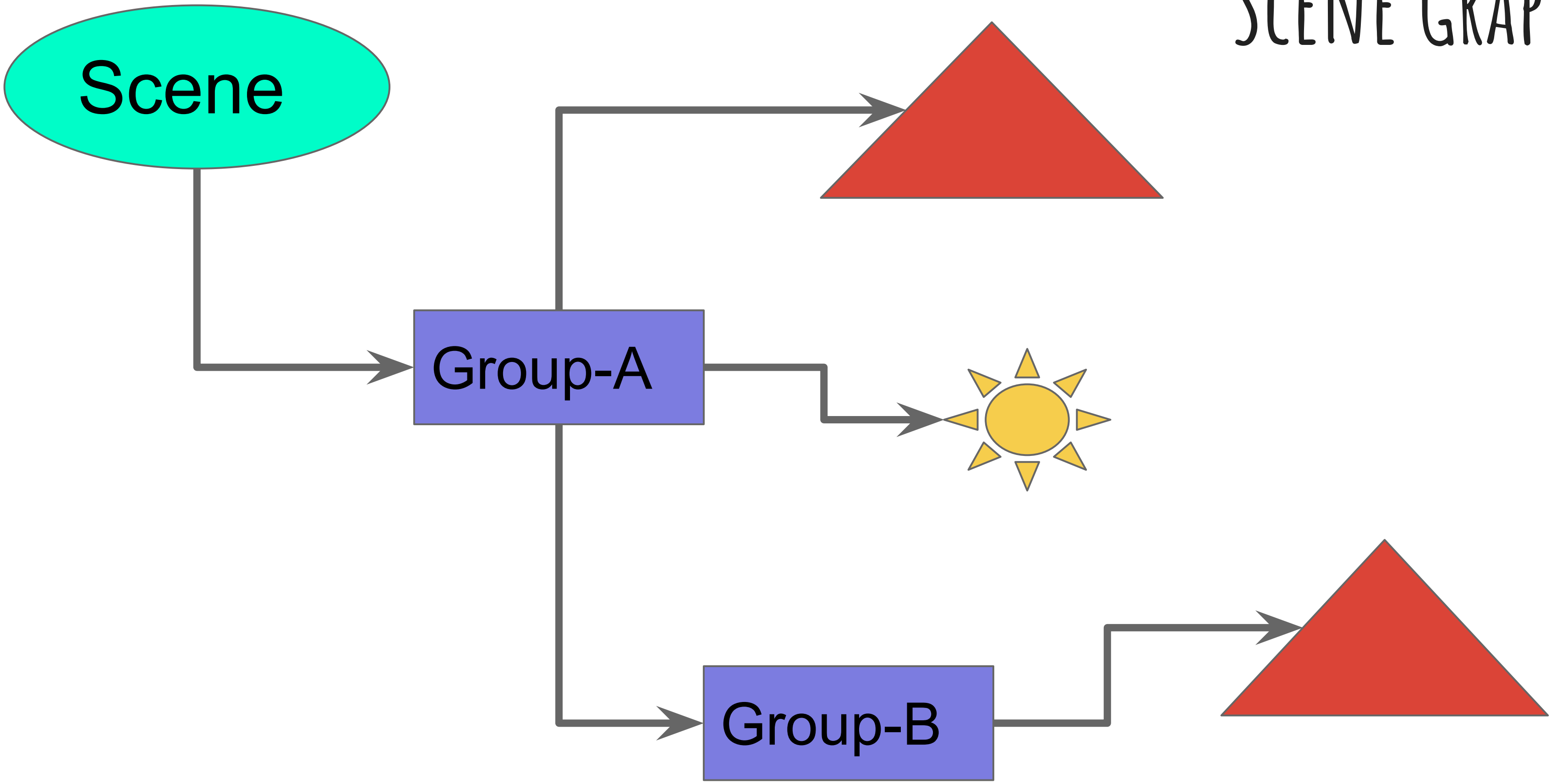
obj-rs

mint

cgmath

collision

# Scene Graph



# Snippets

Window creation:

```
let mut win = three::Window::new("Title", "data/shaders");
```

Texture and OBJ loading:

```
let map = win.factory.load_texture("pikachu_anim.png");

let (groups, meshes) = win.factory.load_obj(&path);
```

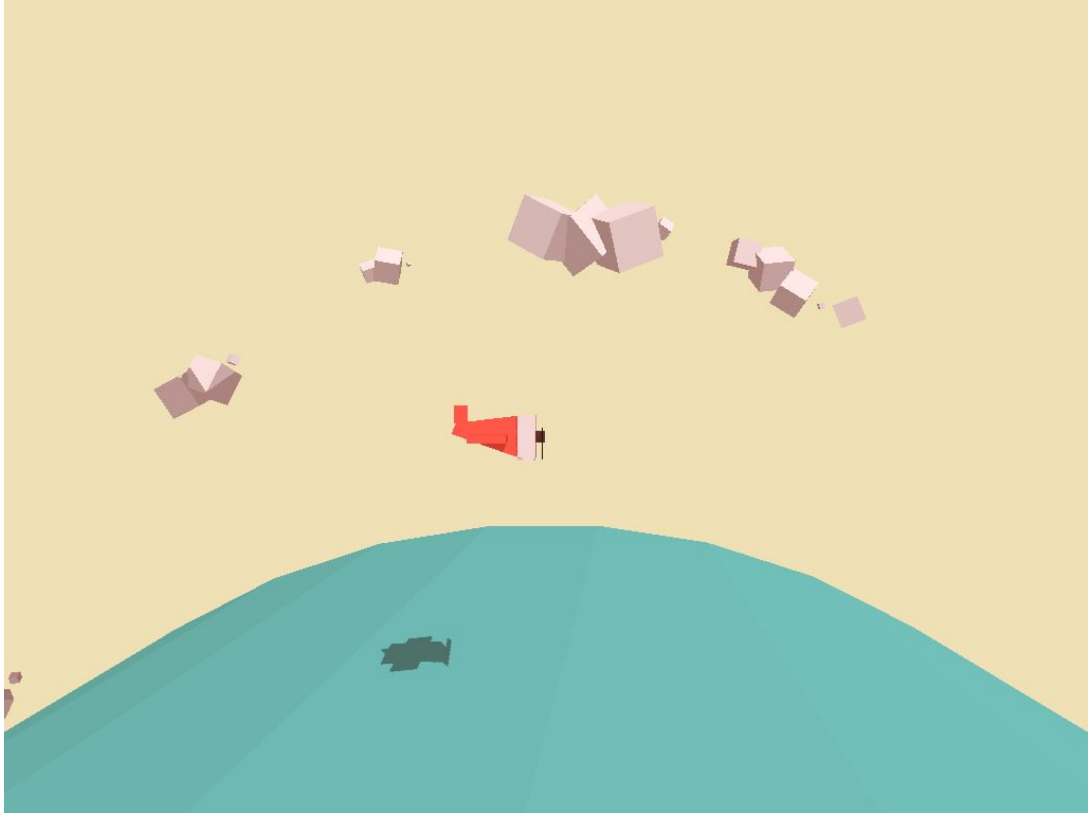
Orbital camera:

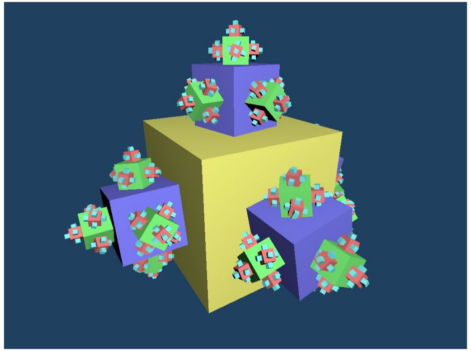
```
let mut controls = three::OrbitControls::new(&cam, [0.0,
2.0, -5.0], [0.0, 0.0, 0.0]);
```

# Current state

- *Basic shapes*: box, cylinder, plane, lines
- *Lights*: hemisphere, directional, omni, spot
- *Materials*: flat, Lambert-lit, Phong-lit
- *Cameras*: ortho/perspective
- Basic input, including orbital controls
- *Loaders*: images, OBJ
- *Animation*: sprite UV, blend shapes

# Demo time!





# Where are we in 5 years?

- Animations
- Instancing
- Physics and ray casting
- Custom materials
- More demos!

# Links

Code:
https://github.com/kvark/three-rs

Chat:
https://gitter.im/three-rs/Lobby

Slides:
https://github.com/kvark/slides/blob/master/ThreeRS_RustTorontoMeetup.pdf