Advancing Web Gaming To New Heights with Cocos

Huabin LING (aka panda)
Cocos Technical Director

Outline

- Cocos Creator introduction with game cases
- Engine architecture on Web and core features
- Our journey to WebGPU

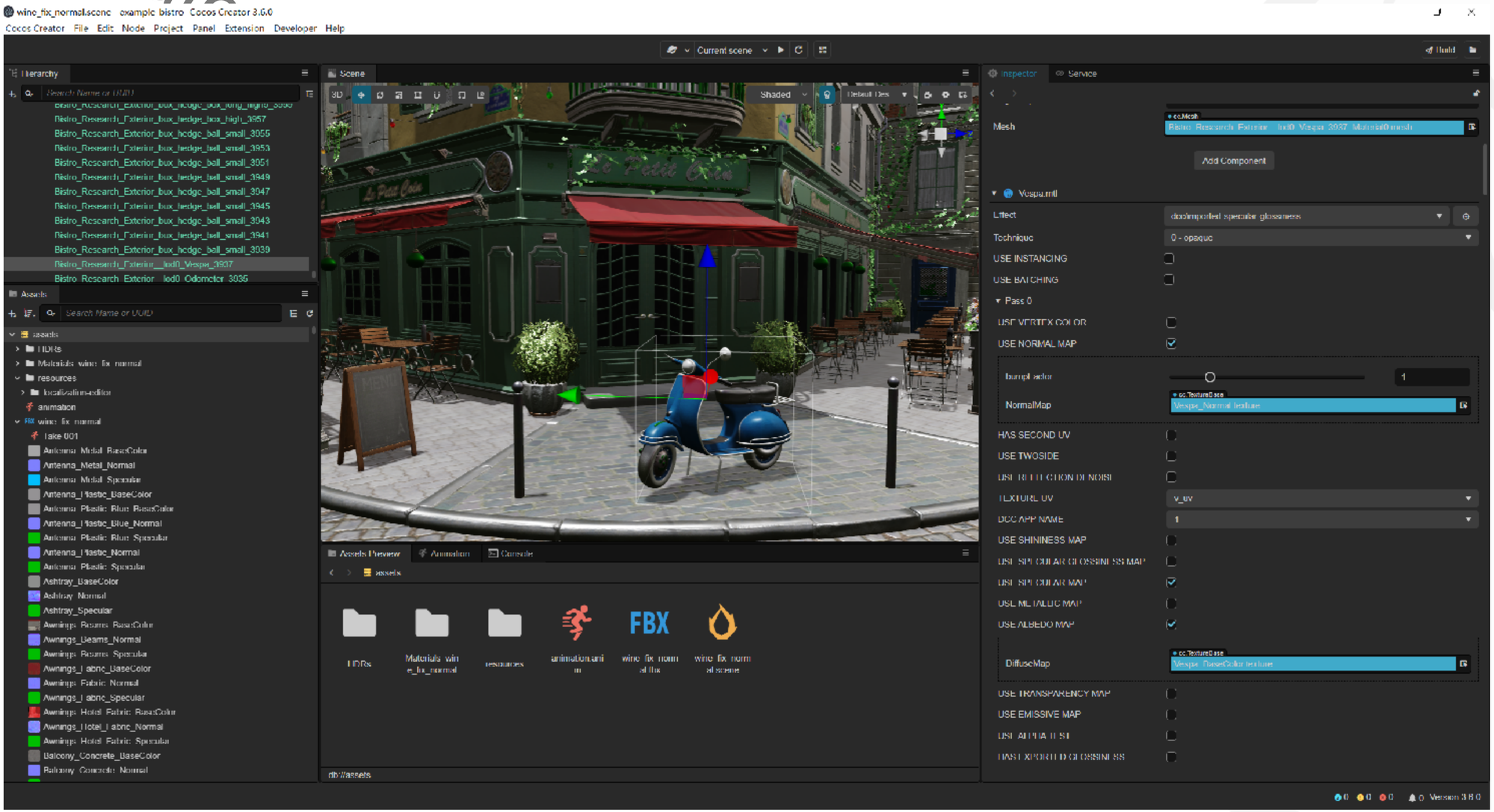
01

02

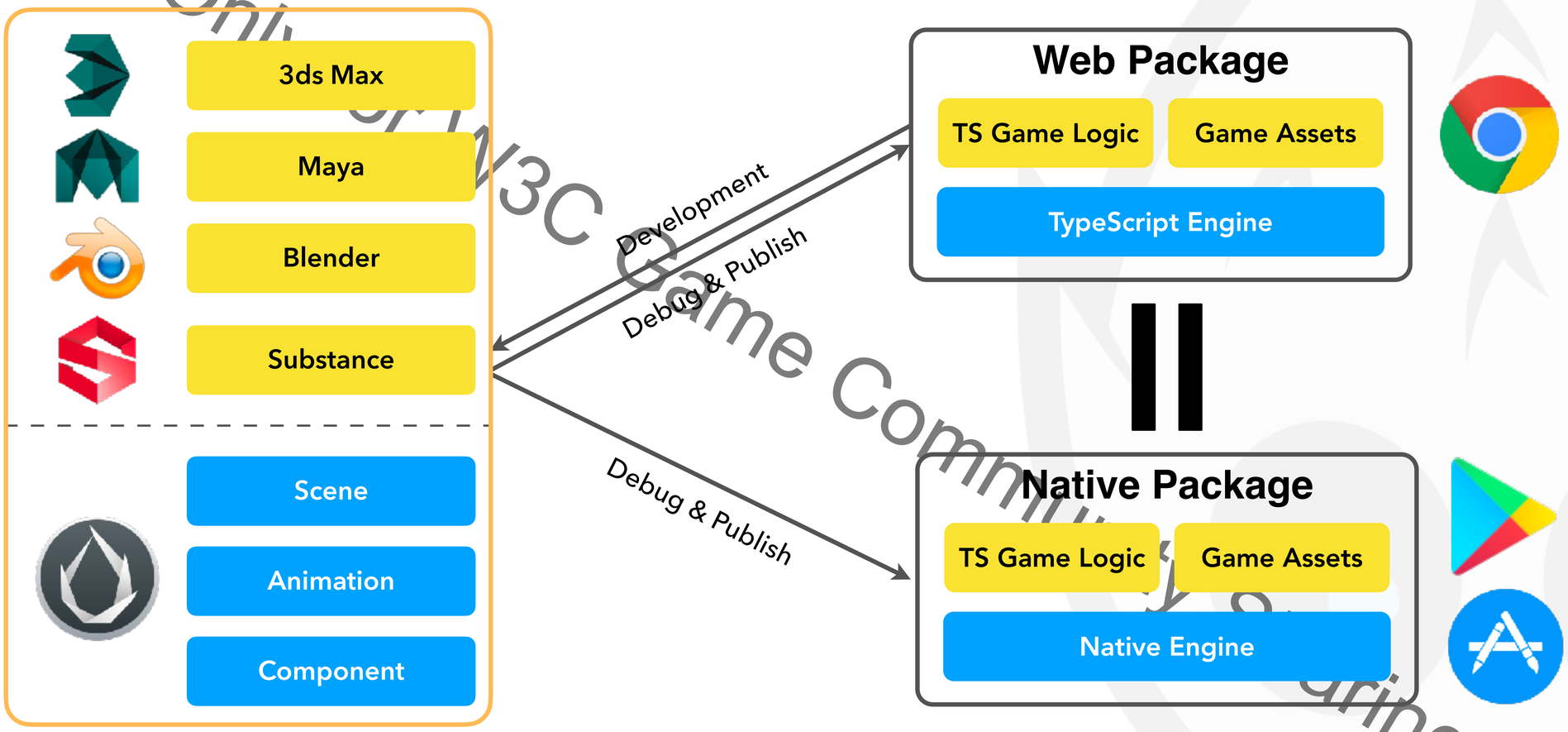
03

Cocos Creator Introduction

Cocos Creator



Efficient Web Based Workflow



Top War

- Top ranking game
- Platform: Web + Mobile Native
- Merge + 4X SLG
- Merge gameplay makes ads buying much more efficient
- 100 millions downloads
- $20m monthly revenue

Ubisoft Nano

- 10 multiplayer party games on web
- 21 Ubisoft IP landed on web
- 10 million players (2021)
- 188 countries (2021)
- 8 platforms: Web portals, telecom operators, social Apps
- 10mb package size, 4s loading time

Three Kingdoms

- Released in Korea and Taiwan
- Platform: Google Play, Apple Store, Web
- Highly unified experience
- 4X SLG

Complex Render Pipeline on Web

https://preview.cocos.com/lake/

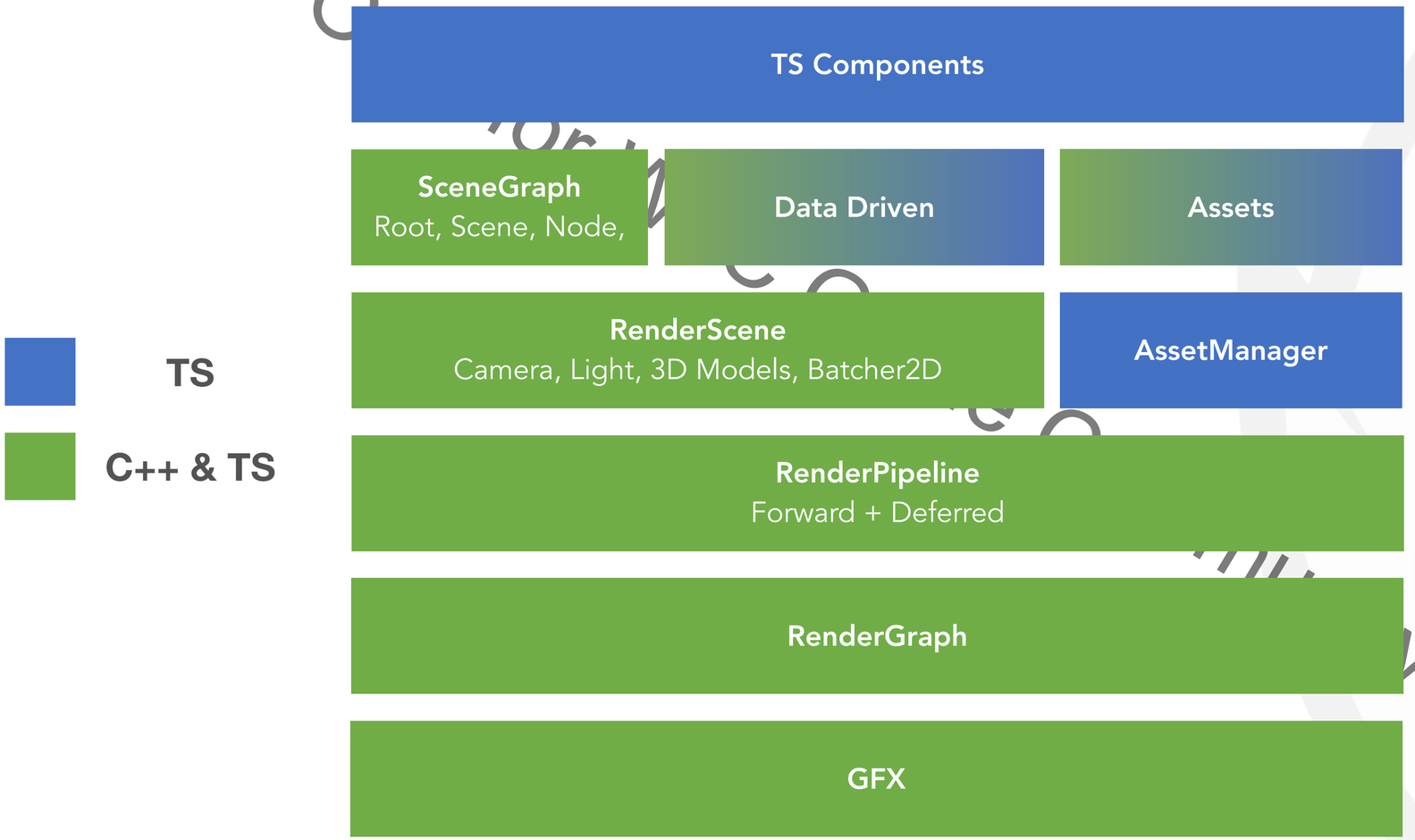
01

02

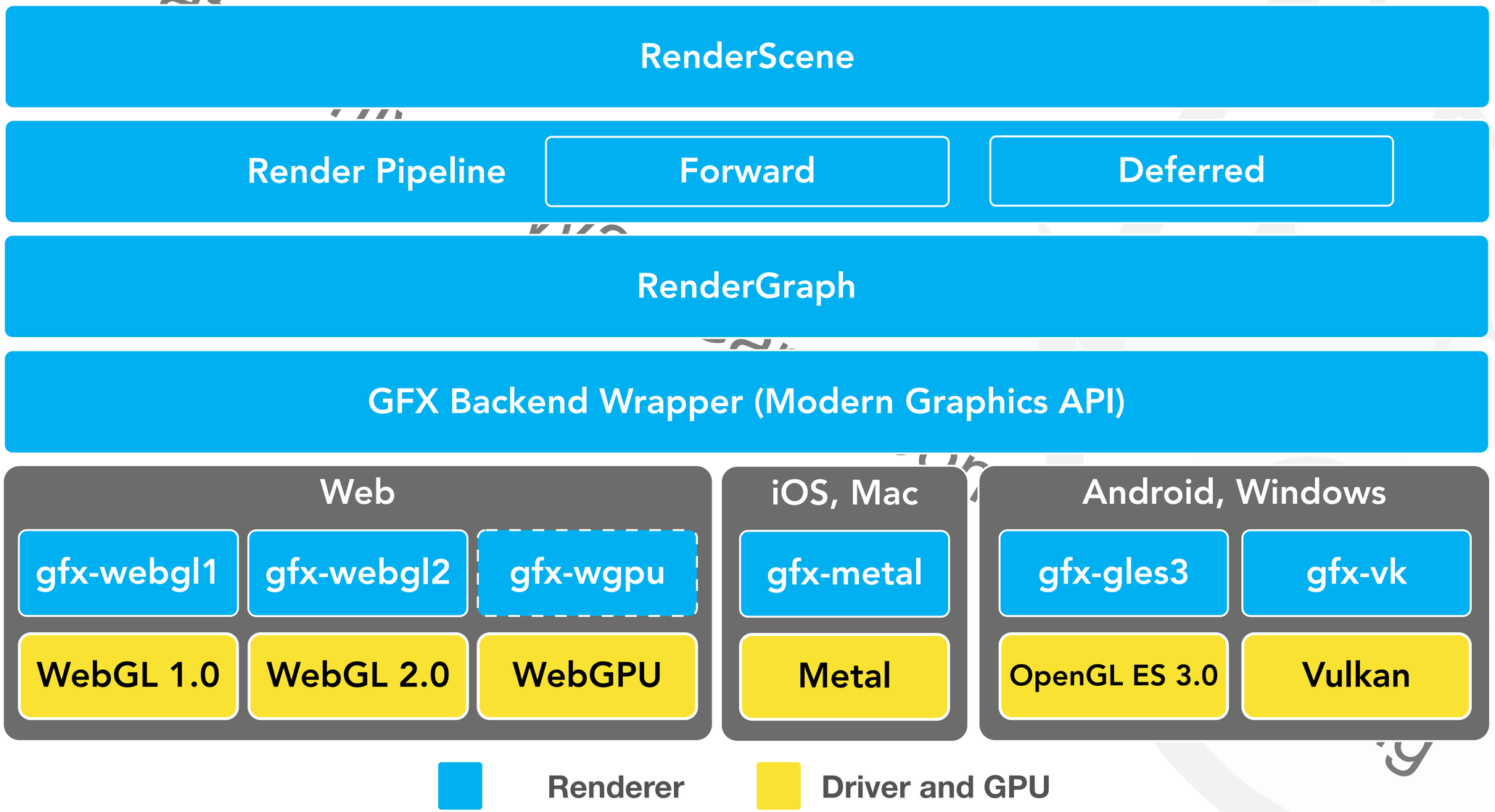
03

Engine Architecture and Features

Engine Architecture

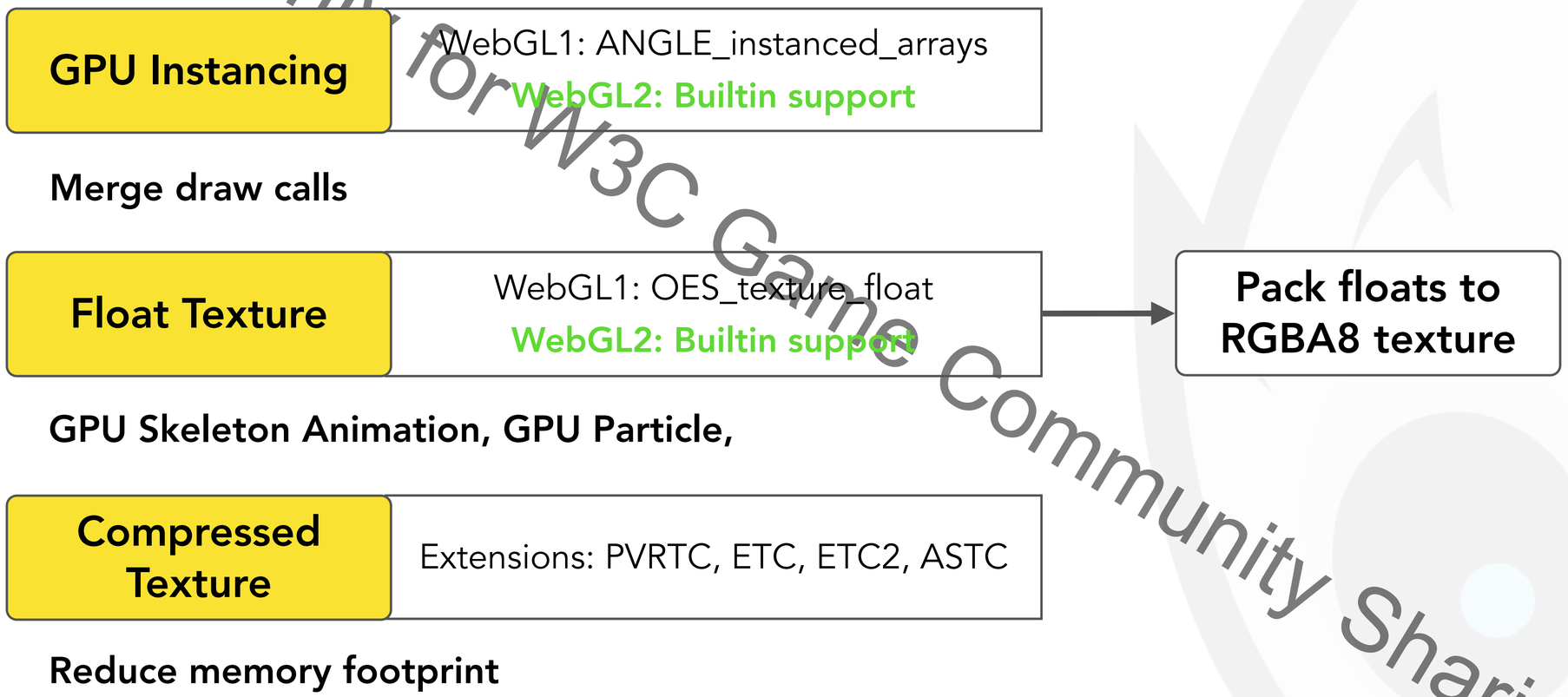


Cross Platform Renderer



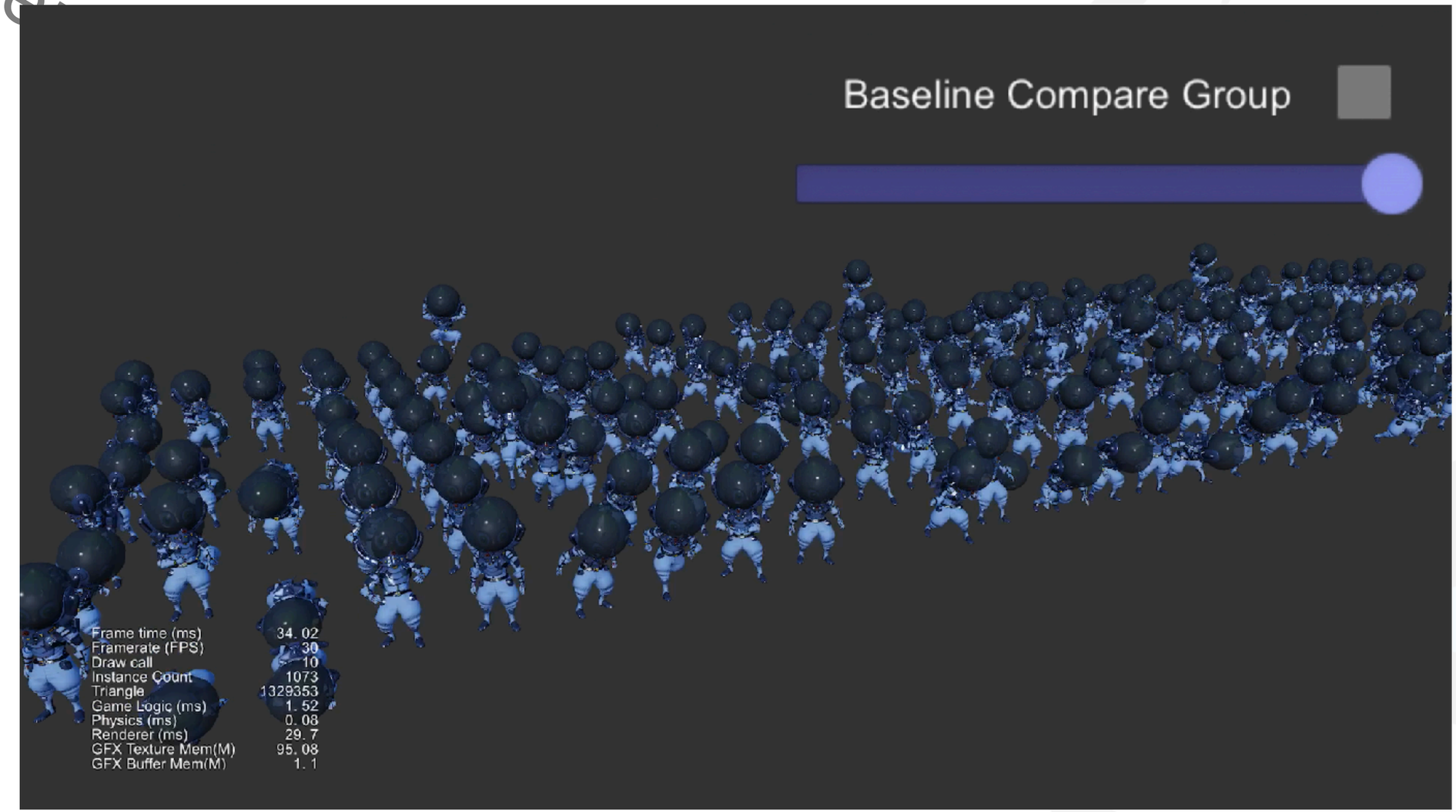
Graphics Backend Upward Adaptation

Feature Backward Compatibility for WebGL

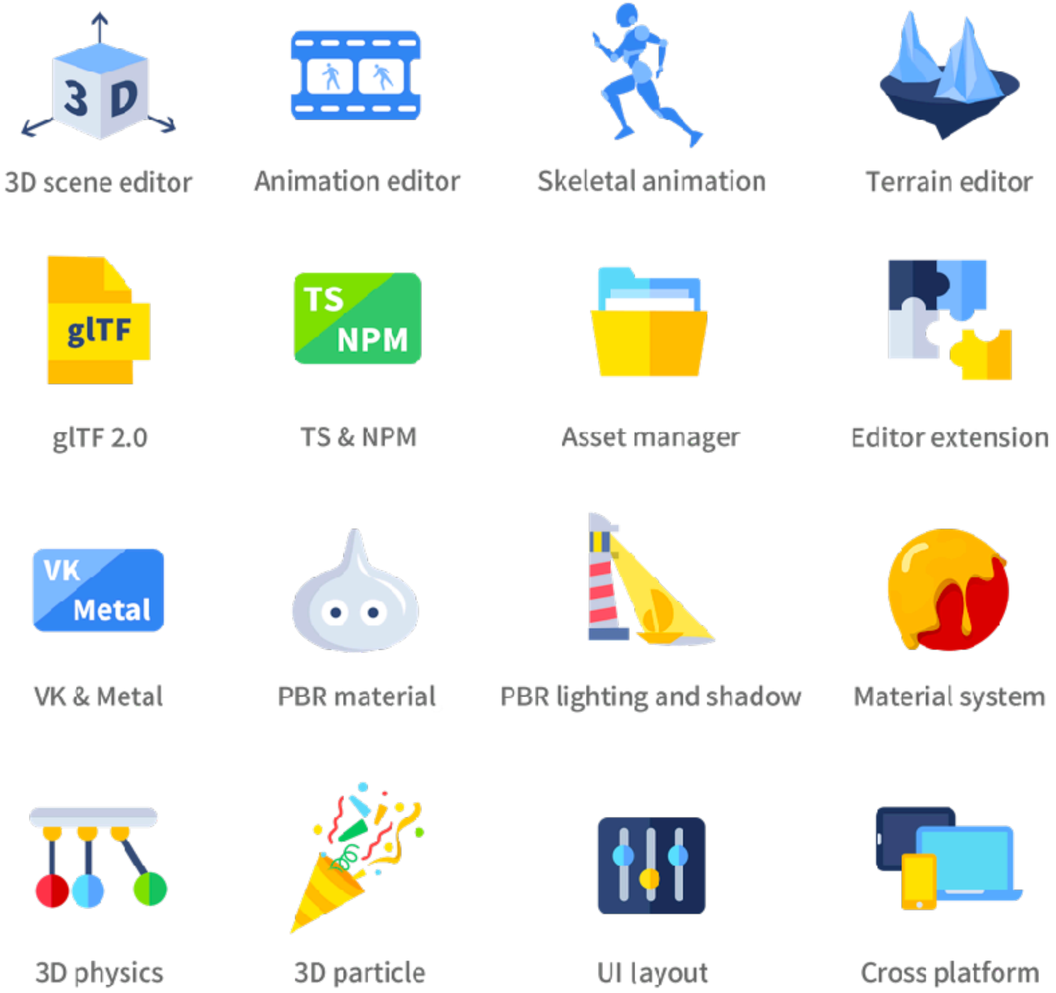


Outcome: GPU Based Skeletal Animation

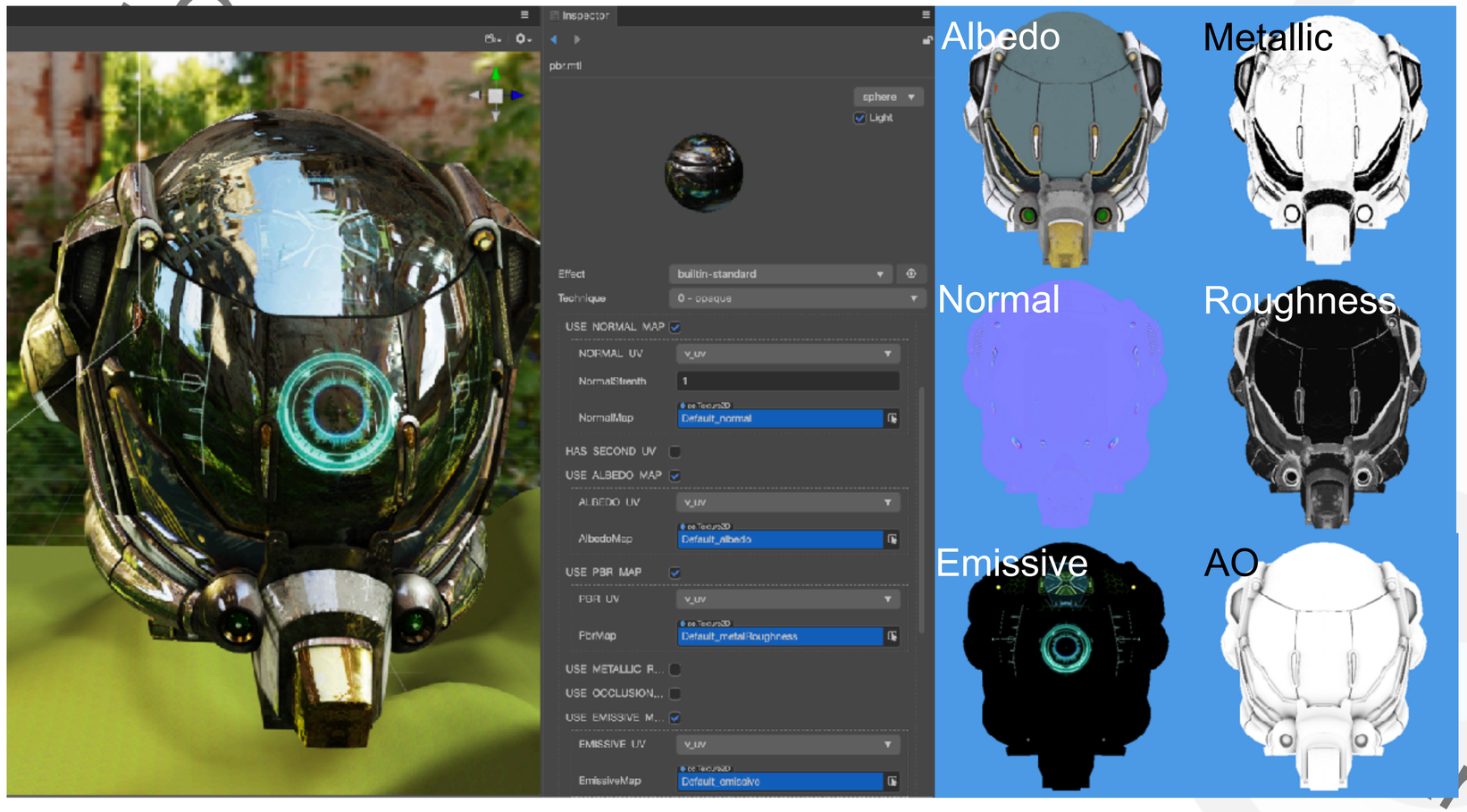
iPhone 7 plus
WebGL1
Full PBR model
Instances: 1000
Draw Call: 10
FPS: 30



Engine features and recent updates



Physically Based Rendering

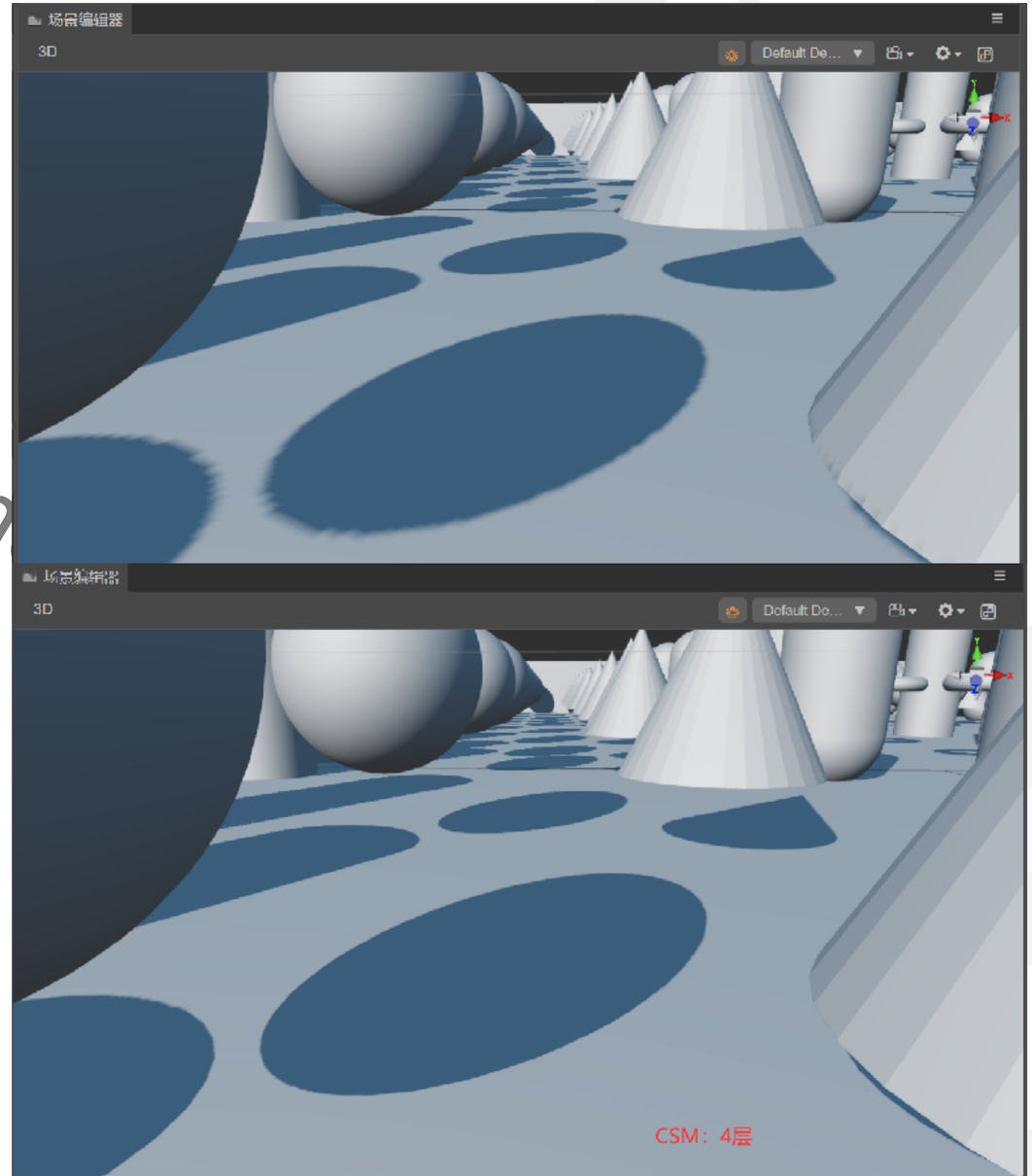


Lighting

- Physically based metrics
- Multi-pass direct light
- Diffuse with environment convolution map
- Reflection with GGX convolution mipmaps
- IBL reflection denoise

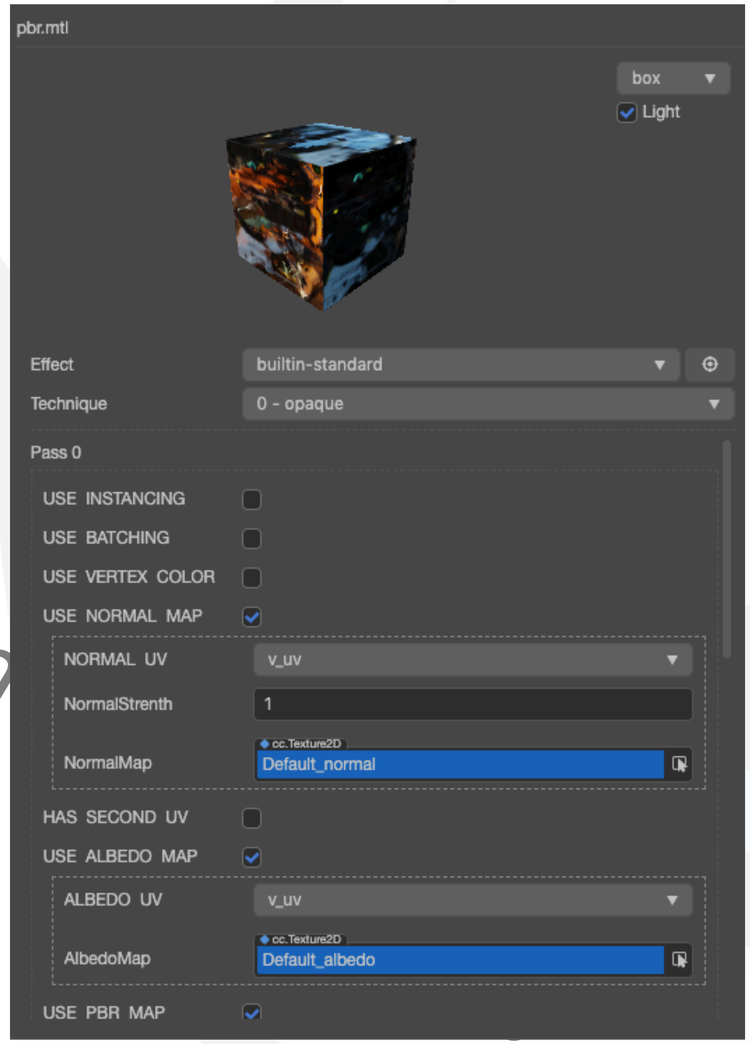
Shadows

- Supports planar shadow and shadow map
- PCF soft shadow
- Depth bias and model bias
- Four layer cascade shadow map
- Support instancing batch



Material System

- Effect based on YAML descriptive format
- Effect with Surface Shader customization
- Shader using GLSL 300 es
- Keep uniform value while switch materials
- Uniform key frame animation support

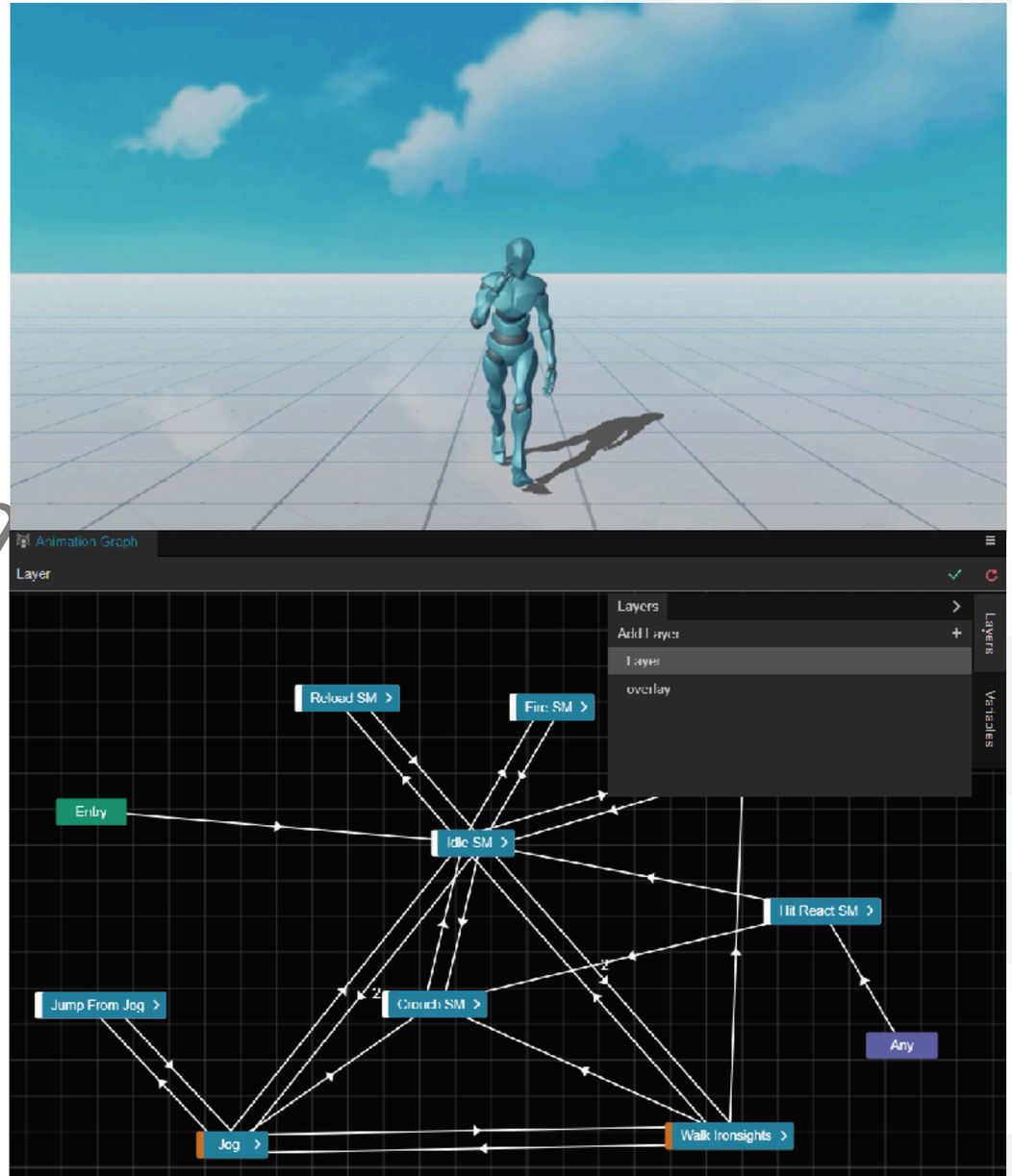


Smart Material Conversion

- Supports 3ds Max, Maya, Blender, C4D
- Supports Lambert, Blinn, Phong, PBR
- Convert traditional lighting model to PBR lighting model
- Supports both Metallic Roughness and Specular Glossiness workflows

Marionette Animation System

- State machine and sub state machine
- Blend Tree
- Event Trigger
- Animation Layer
- Skeleton Masking

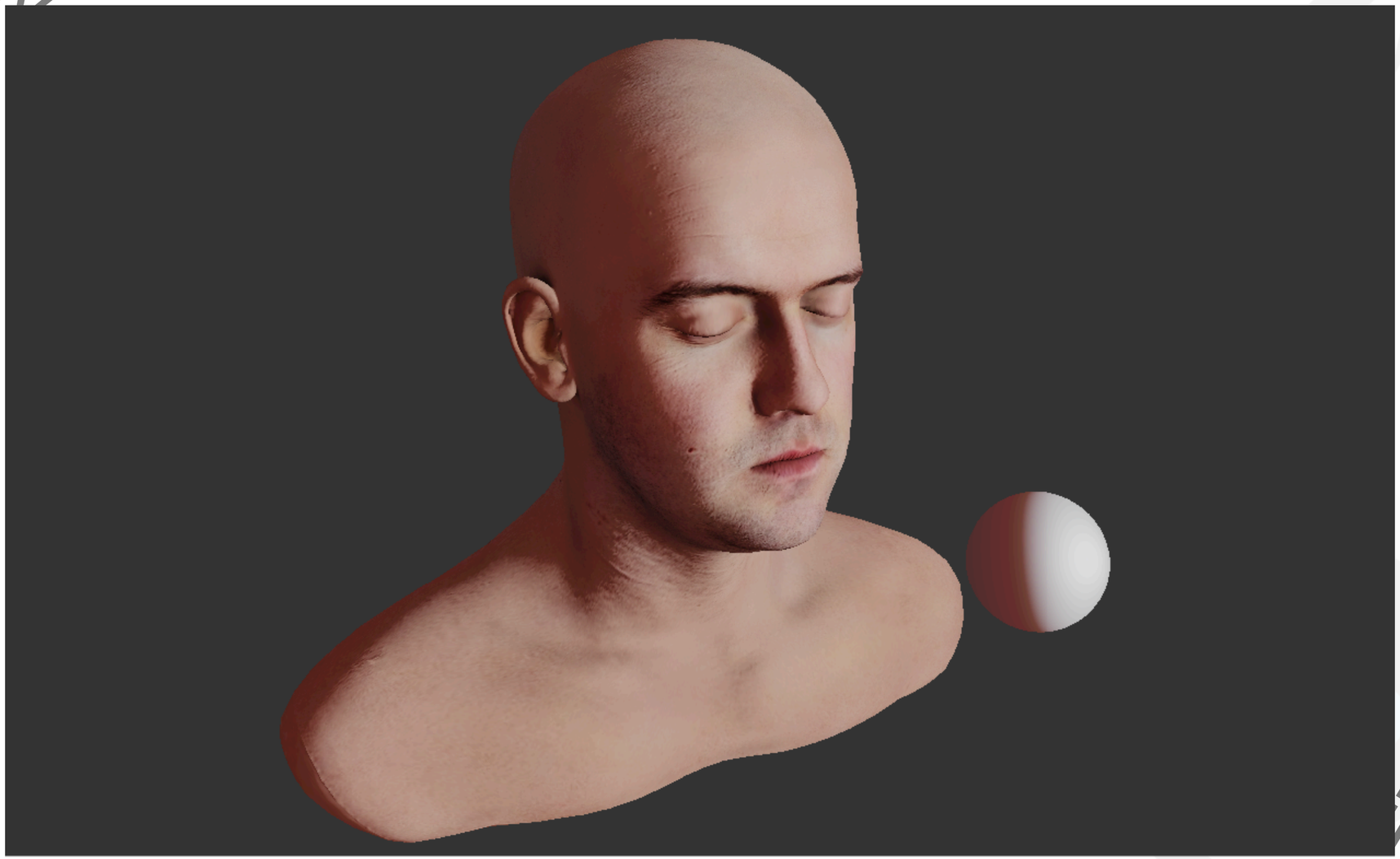


02

03

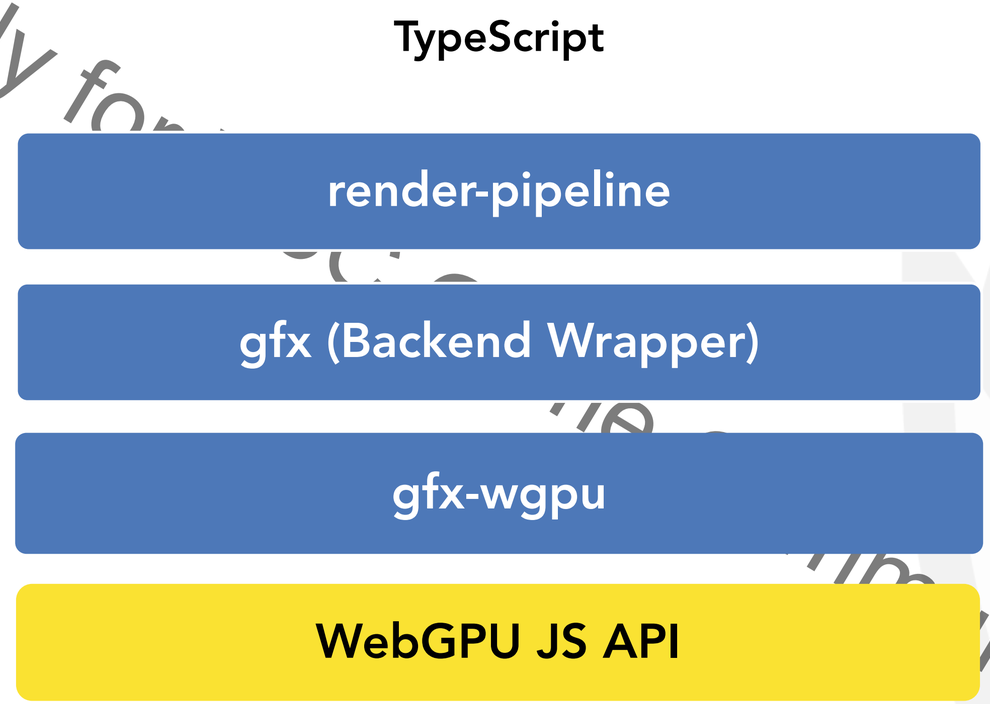
Our Journey to WebGPU

Our first WebGPU demo

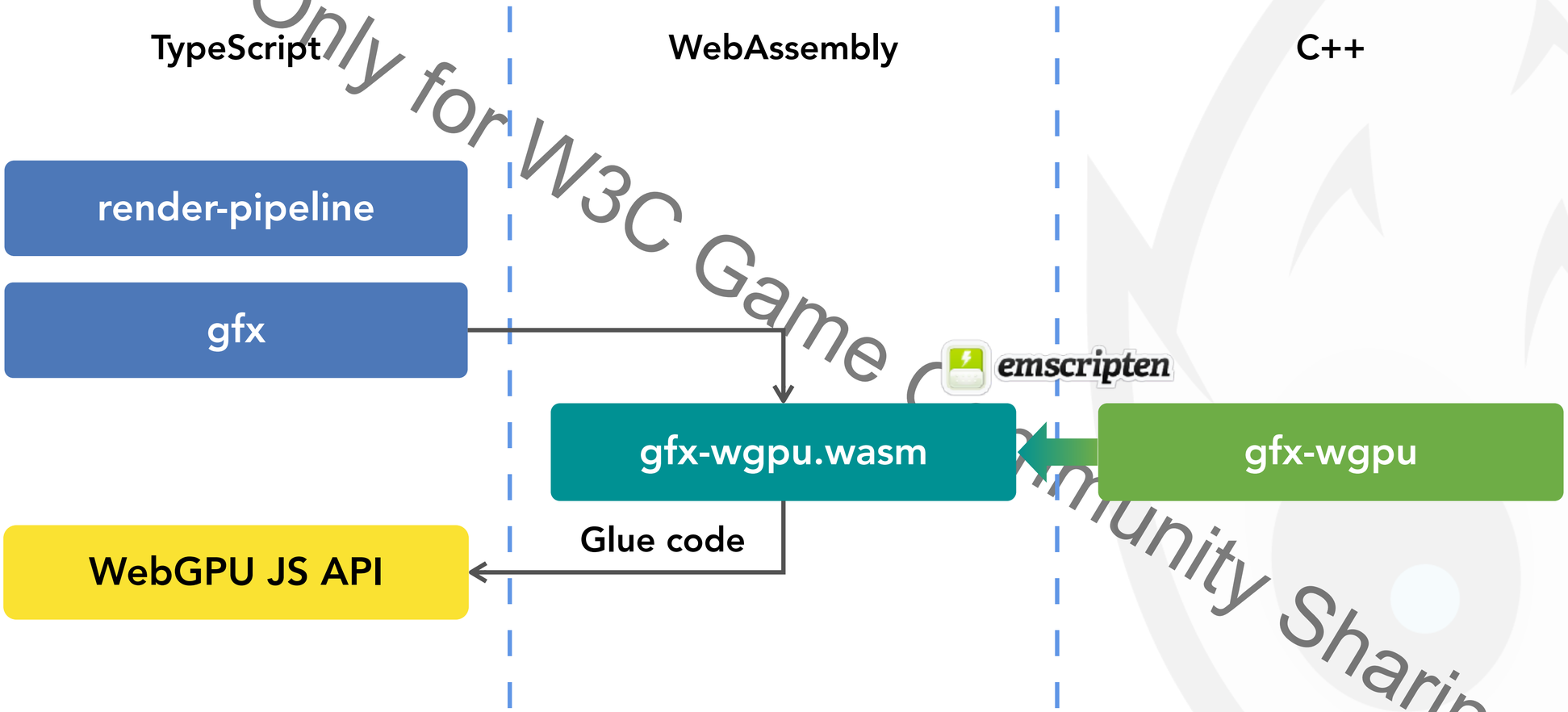


Chrome Canary 900323: https://game.cocosjoy.com/games-preview/web-desktop/

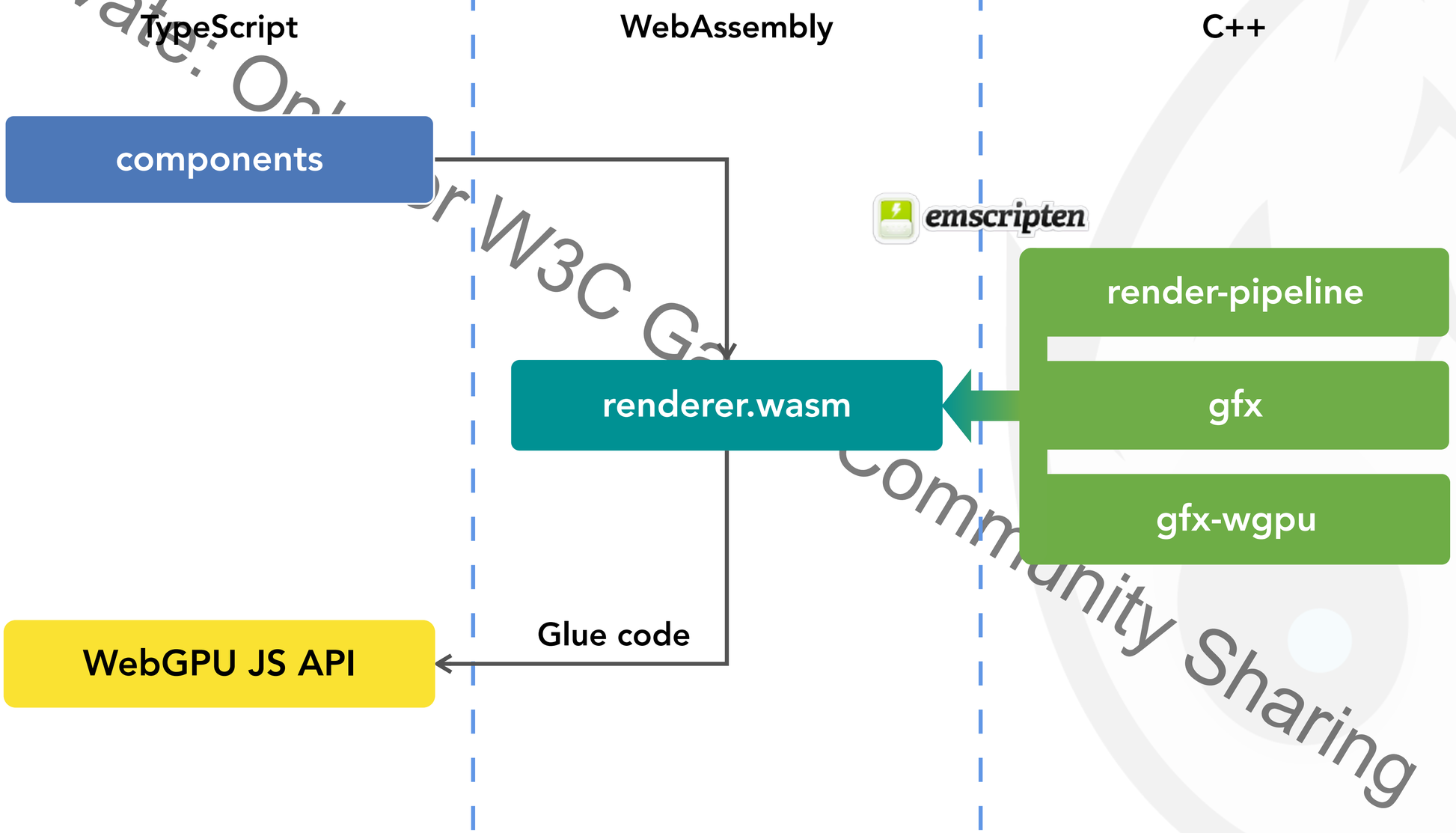
Initial implementation: The classic web way



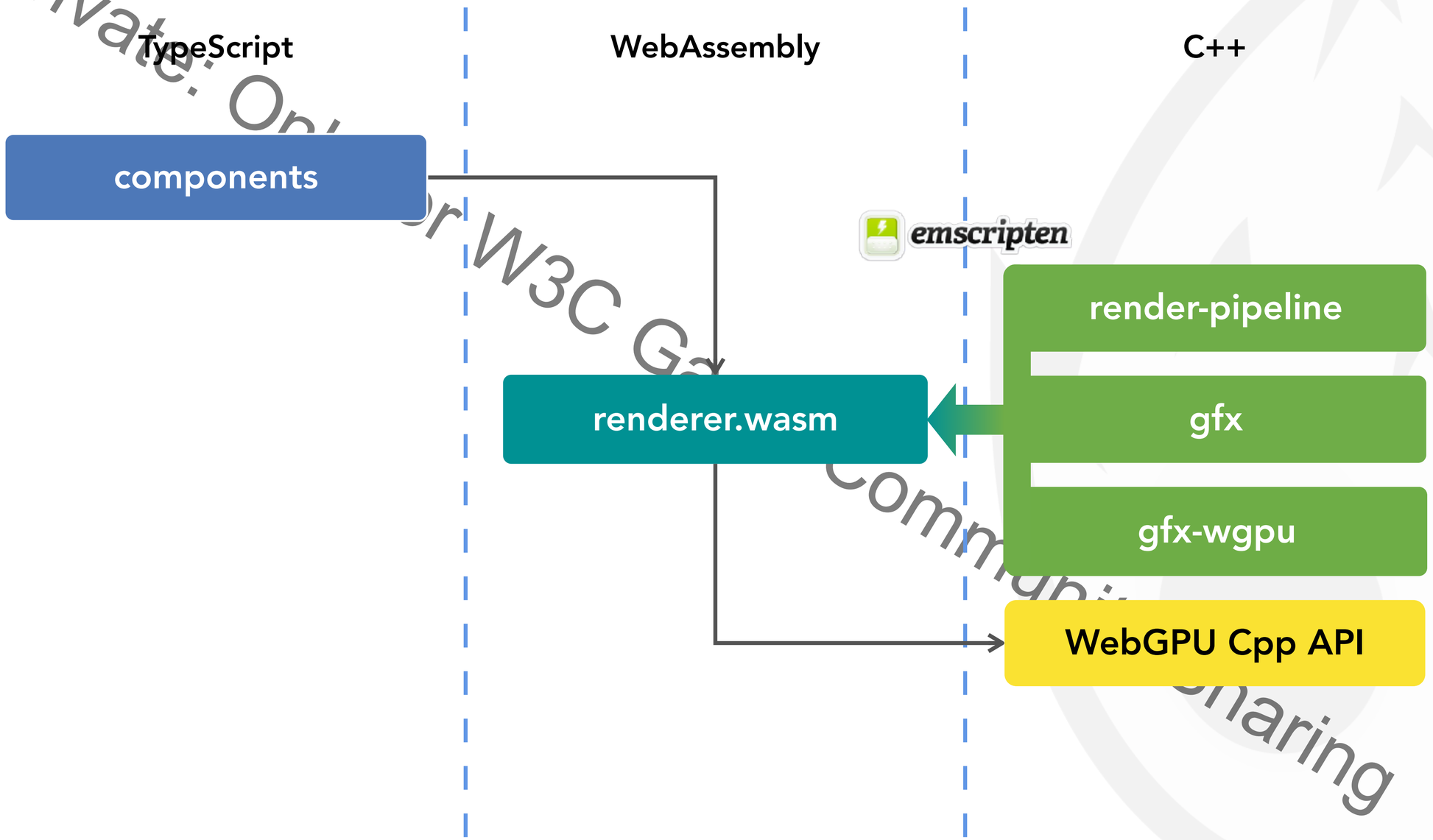
Current implementation: Native WebGPU



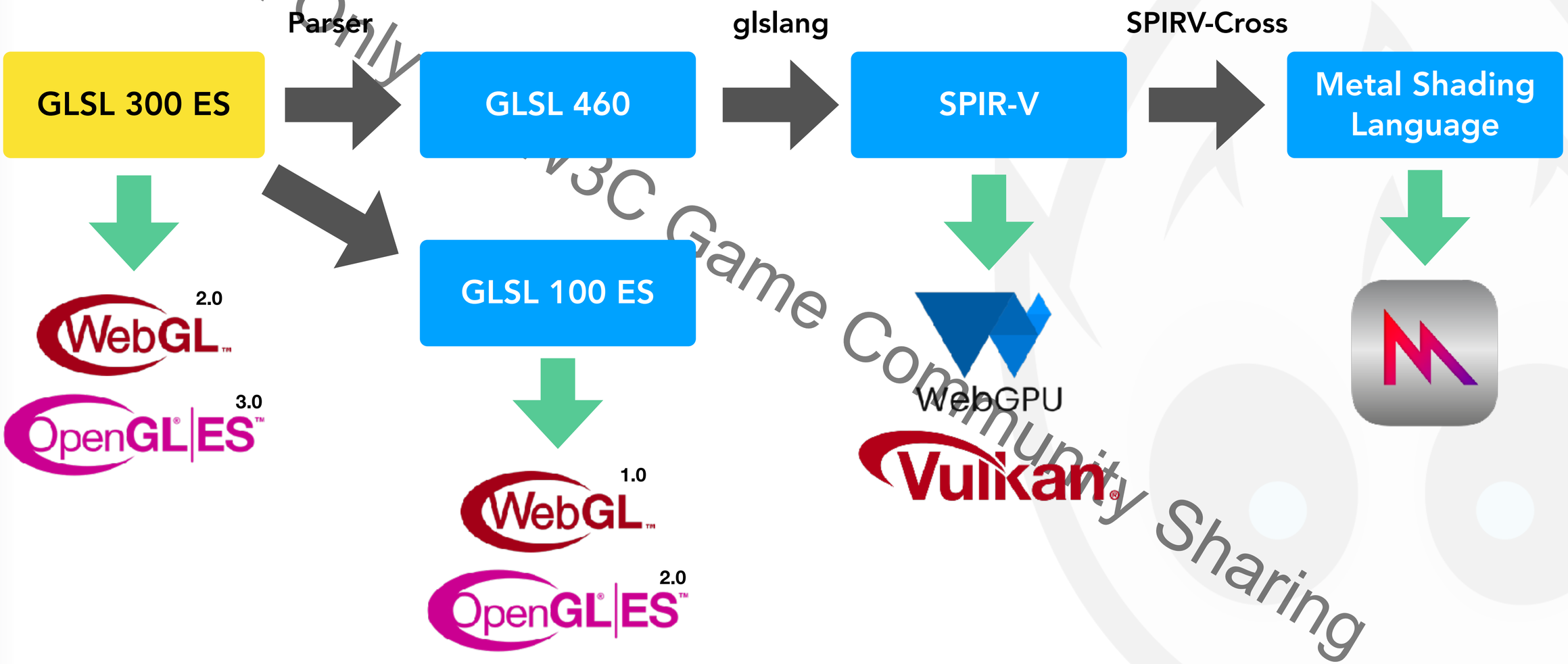
Next step



Hmmm, why not ? 🤔


TypeScript
WebAssembly
C++
components
emscripten
renderer.wasm
render-pipeline
gfx
gfx-wgpu
WebGPU Cpp API

Full shader compilation graph



Design principles of our architecture

- Performance is key !
- Make cross platform development easy and as seamless as possible.
- Maximize accessibility for our game.
- Build future ready infrastructure.
- Bring next generation graphics and computing power to the Web.
- Reduce engine maintenance cost.

Stay tuned for new updates

THANKS

Social medias: @cocosengine

panda@cocos.com