


GAMES 106

现代图形绘制流水线原理与实践

霍宇驰
eehyc0@gmail.com

L1 2023/4/5

课程团队

老师

霍宇驰&袁亚振&高希峰
浙江大学　腾讯　北美腾讯光子

胡义伟&高涛
耶鲁大学　相芯科技

2023年4月5日起 | 北京时间每周三下午14:00-15:00 | WEBINAR.GAMES-CN.ORG

霍宇驰
浙大CAD&CG
杭州光线云
之江实验室

图形绘制　计算光学　机器视觉

袁亚振
腾讯游戏

实时绘制　绘制管线

高希峰
北美腾讯光子

几何计算　网格处理

胡义伟
耶鲁大学
Adobe

纹理材质　可微绘制

高涛
相芯科技

作业主程　绘制引擎

课程团队

戴雨欣 浙江农林大学 buttersdyx@gmail.com

姜伯汉 浙江大学 jiangbohan314@163.com

梁任冬 太极图形 admin@penguinliong.moe

刘紫檀 中国科学技术大学 jauntyliu@mail.ustc.edu.cn

张源娣 上海交通大学 Zydiiii@outlook.com

陈文博 中国科学技术大学 chaf@mail.ustc.edu.cn

王怡贤 华中科技大学 anastasiawangyx@gmail.com

助教同学

GAMES106: https://zju-rendering.github.io/games106/

GAMES106讲什么？

L1 2023/4/5

应用场景

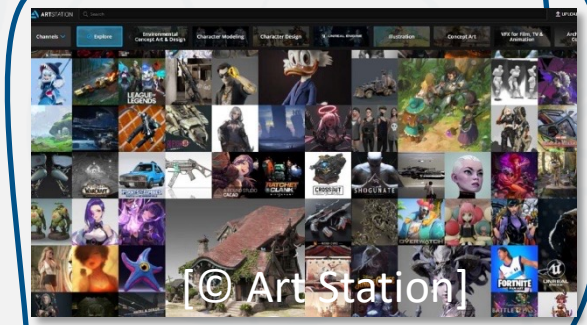
应用场景

设计

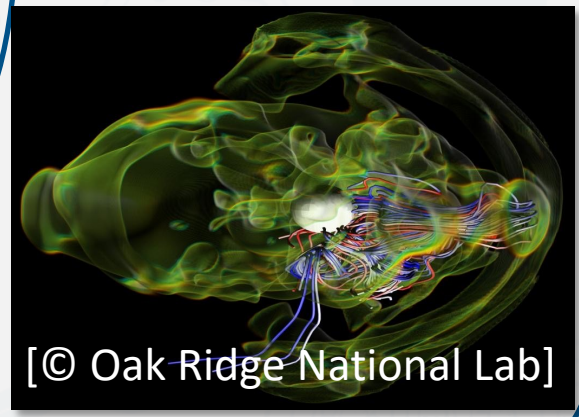
影视

游戏

艺术



可视化



图形绘制

绘制引擎

应用场景

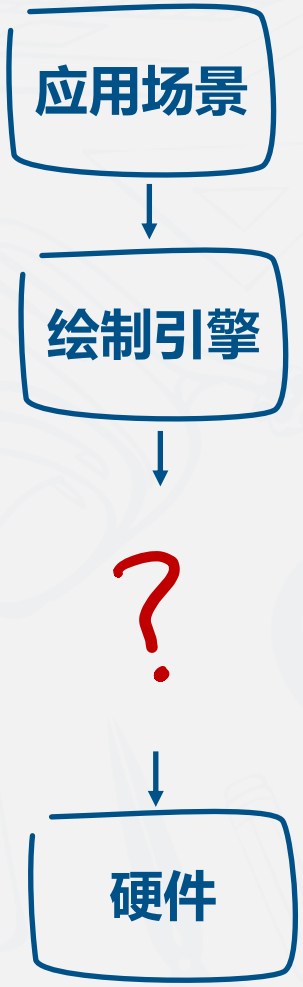
↓

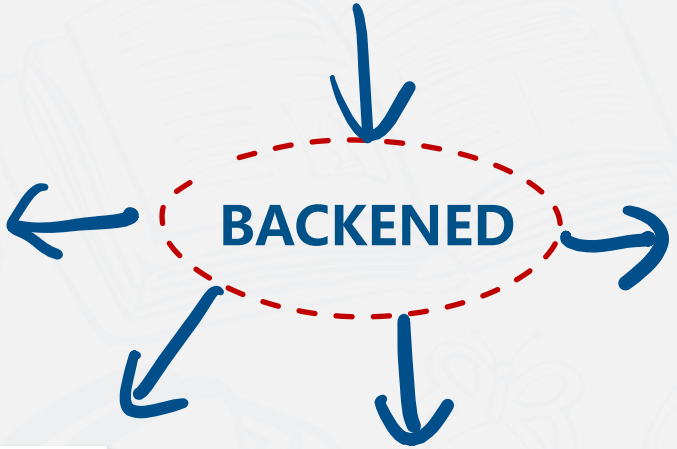
绘制引擎

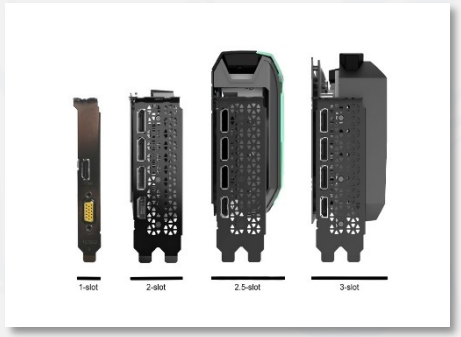
NVIDIA mental ray
NVIDIA Iray
3Ds Max Scanline
ENSCAPE
V-Ray
revizto
unity
UNREAL ENGINE
GODOT Game engine
CRYENGINE
GameMaker

图形绘制

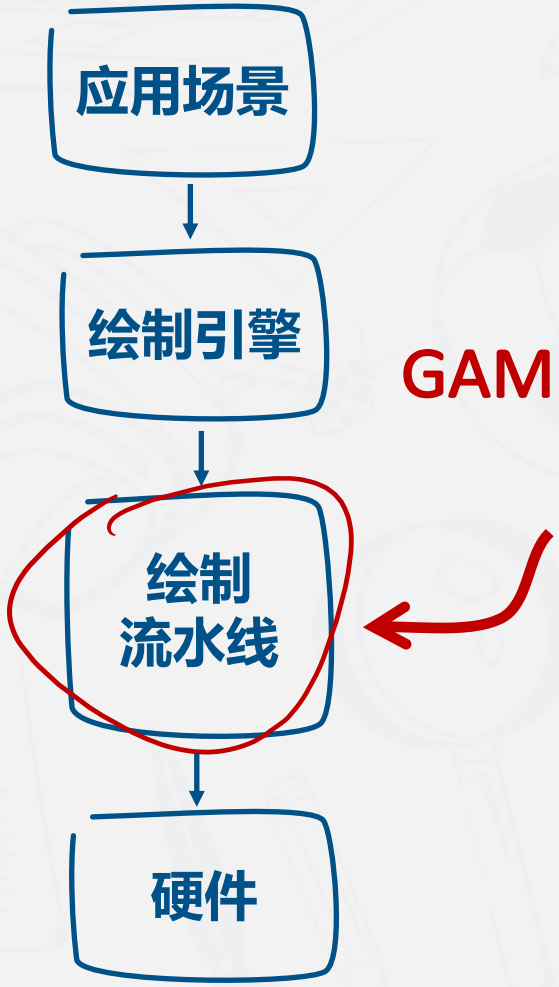
硬件



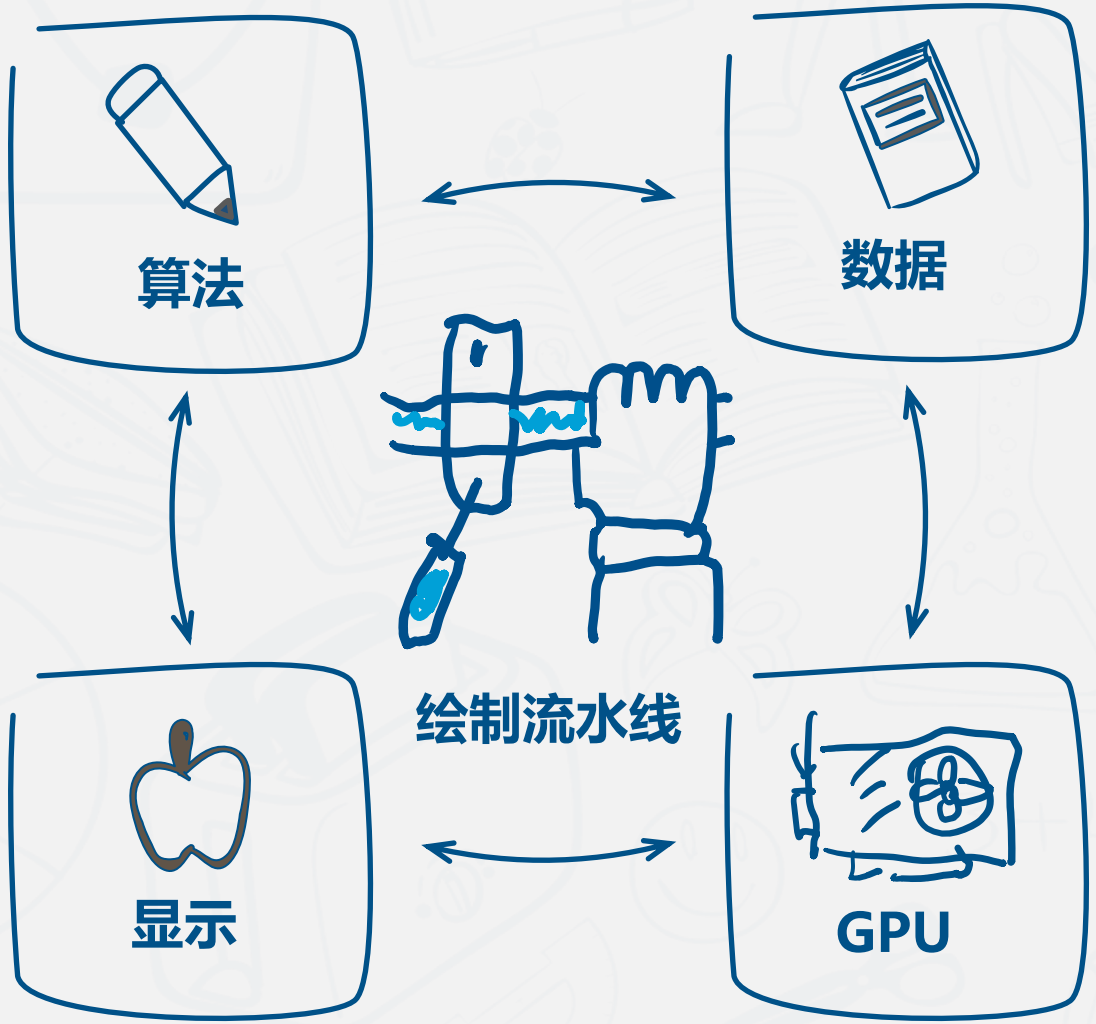




绘制流水线



GAMES 106



图形绘制

绘制流水线


应用场景
绘制引擎
绘制
流水线
硬件
GAMES 106
海量应用
大量引擎
BACKENED
若干
流水线
标准
API
众多硬件（驱动）

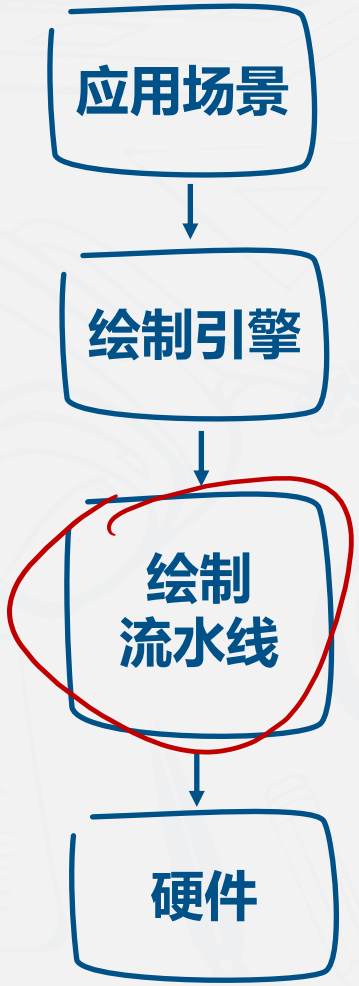
图形绘制

绘制流水线

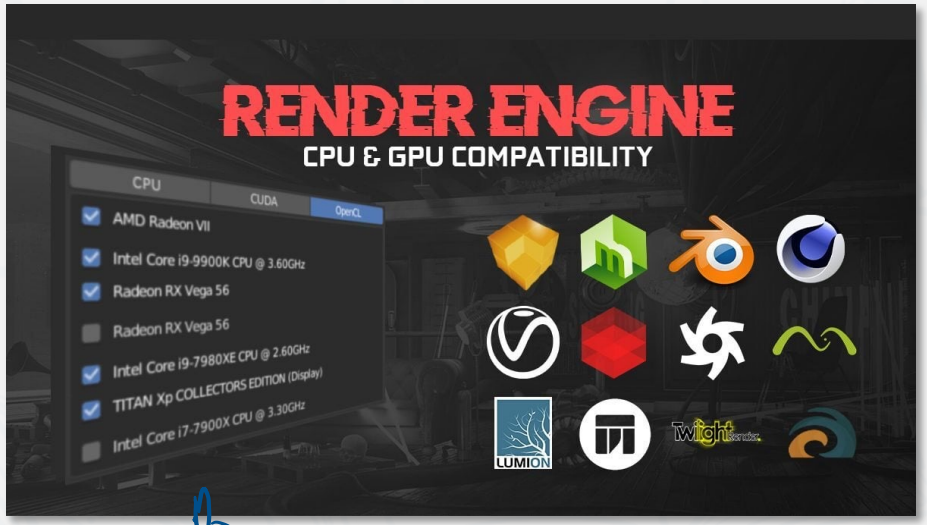

应用场景
绘制引擎
绘制
流水线
硬件
GAMES 106
新算法研发
引擎底层
新特效
程序优化
科研
跨平台兼容
原型开发
中间件
若干
流水线
标准
众多硬件（驱动）

图形绘制

绘制流水线



GAMES
106

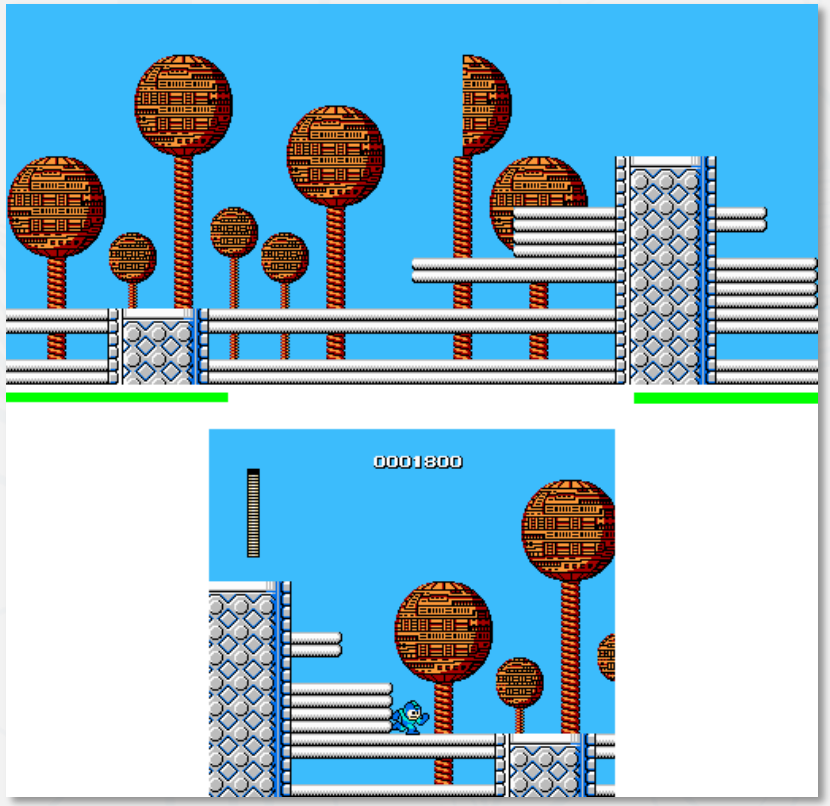


引擎

产业：开发引擎

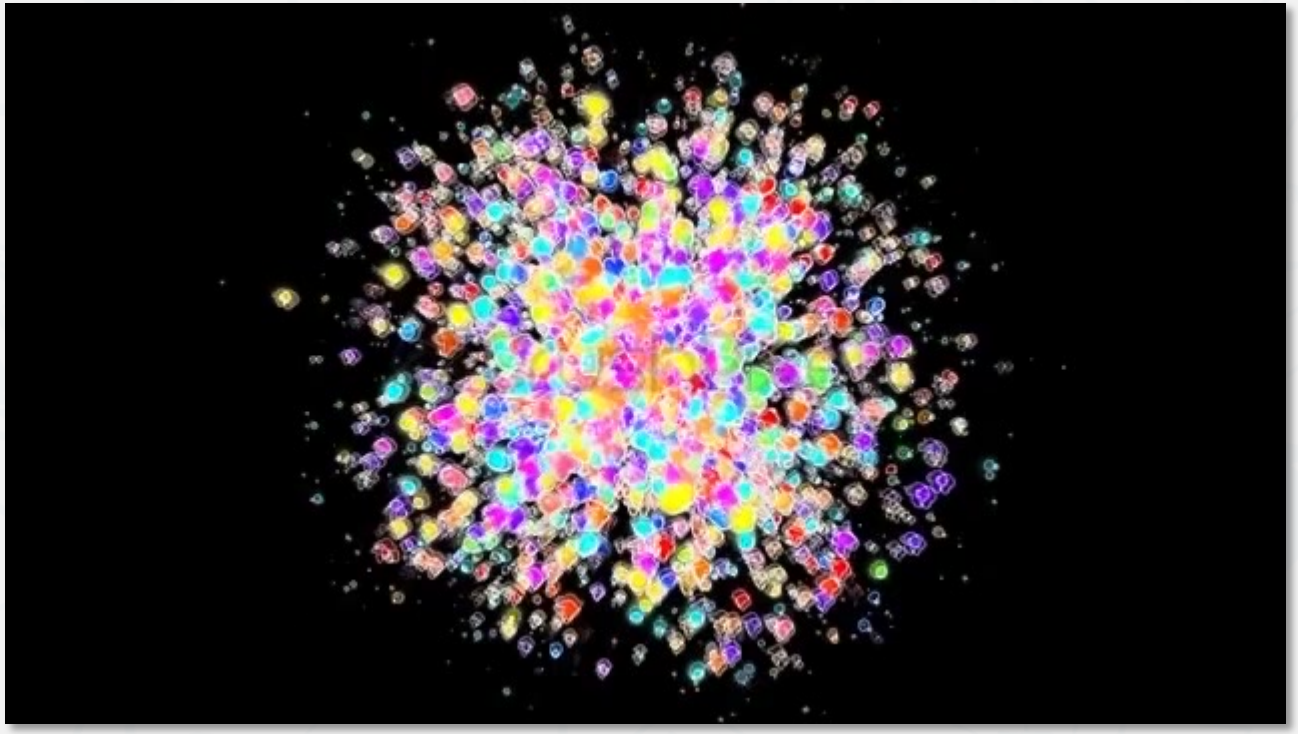
绘制流水线

FC



产业：开发引擎

绘制流水线



产业：实现新特效

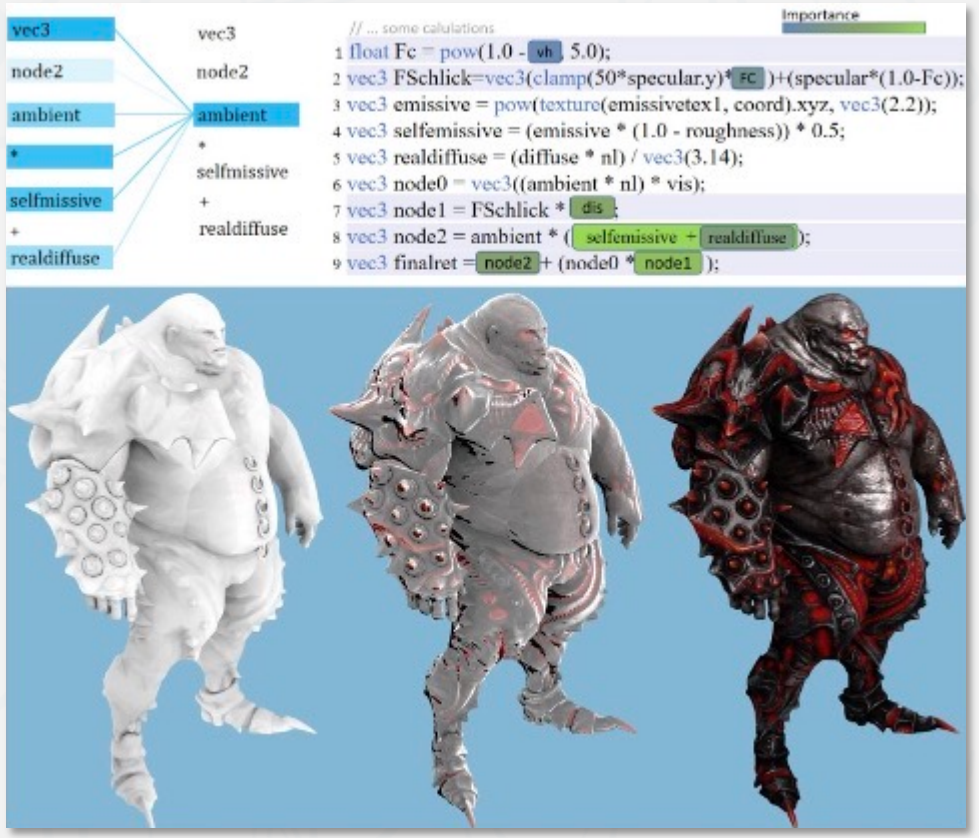
绘制流水线

RTX

产业：实现新特效

图形绘制

绘制流水线



产业：优化图形程序

绘制流水线

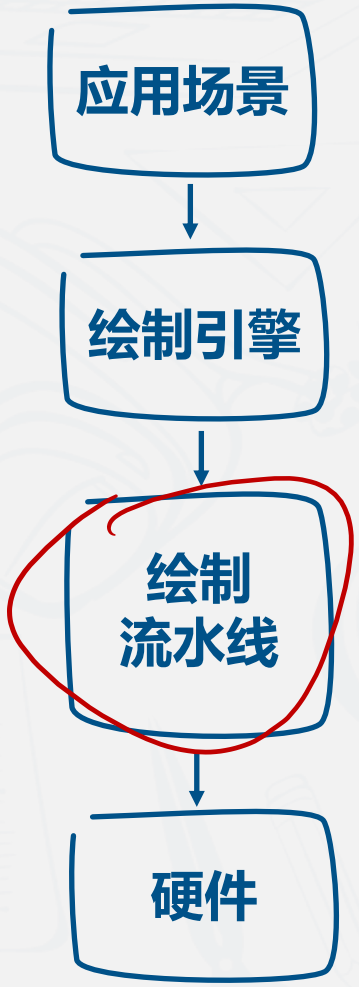
DOOM 3

```
float Q_rsqrt( float number )
{
    long i;
    float x2, y;
    const float threehalfs = 1.5F;

    x2 = number * 0.5F;
    y  = number;
    i  = * ( long * ) &y;                       // evil floating point bit level hacking（邪恶的浮点数位运算黑科技）
    i  = 0x5f3759df - ( i >> 1 );               // what the fuck?（这是什么鬼？）
    y  = * ( float * ) &i;
    y  = y * ( threehalfs - ( x2 * y * y ) );   // 1st iteration （第一次迭代）
//      y  = y * ( threehalfs - ( x2 * y * y ) );   // 2nd iteration, this can be removed（第二次迭代，可以删除）

    return y;
}
```

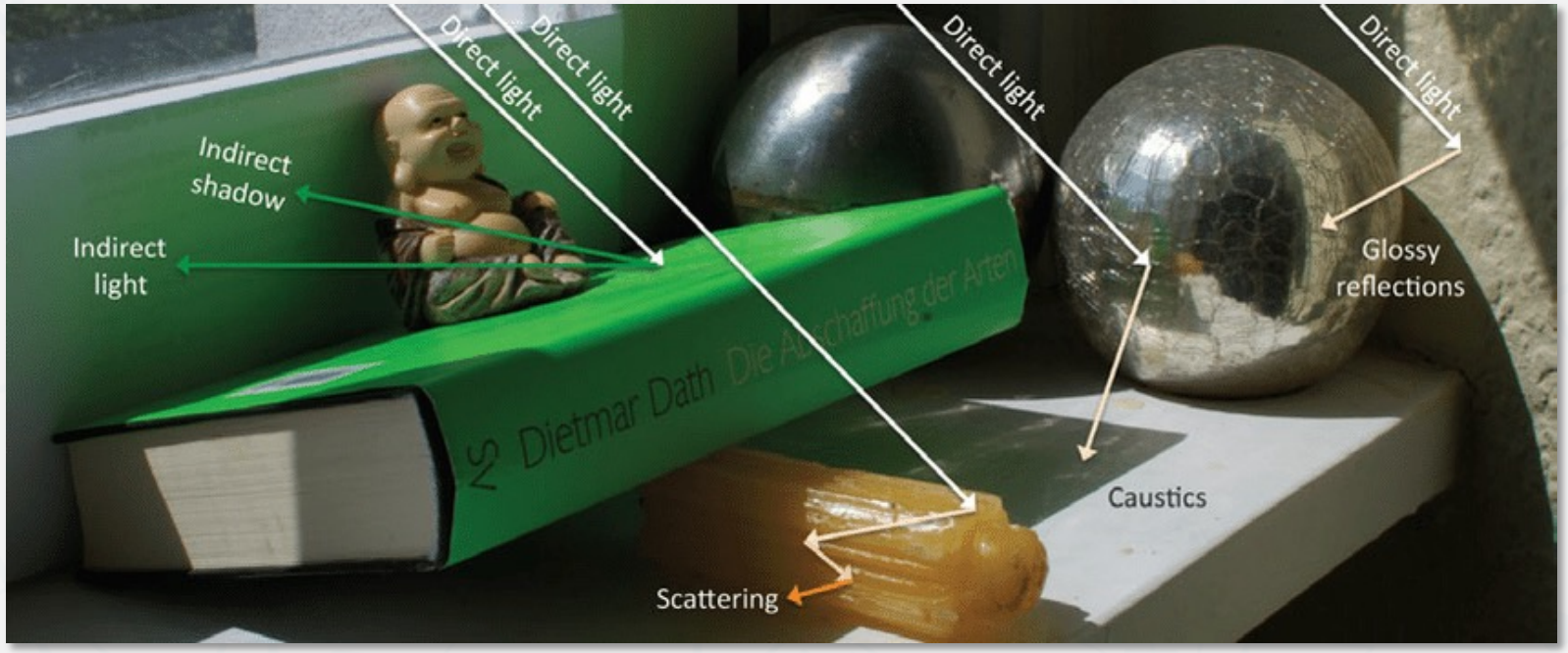
产业：优化图形程序

图形绘制

绘制流水线



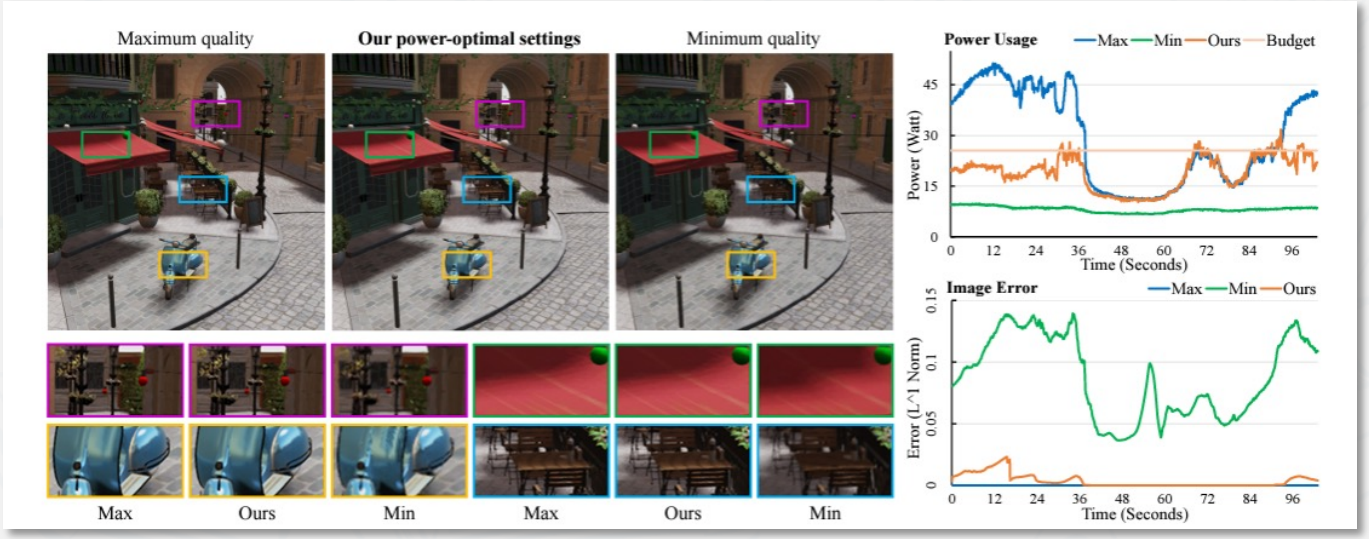
GAMES
106

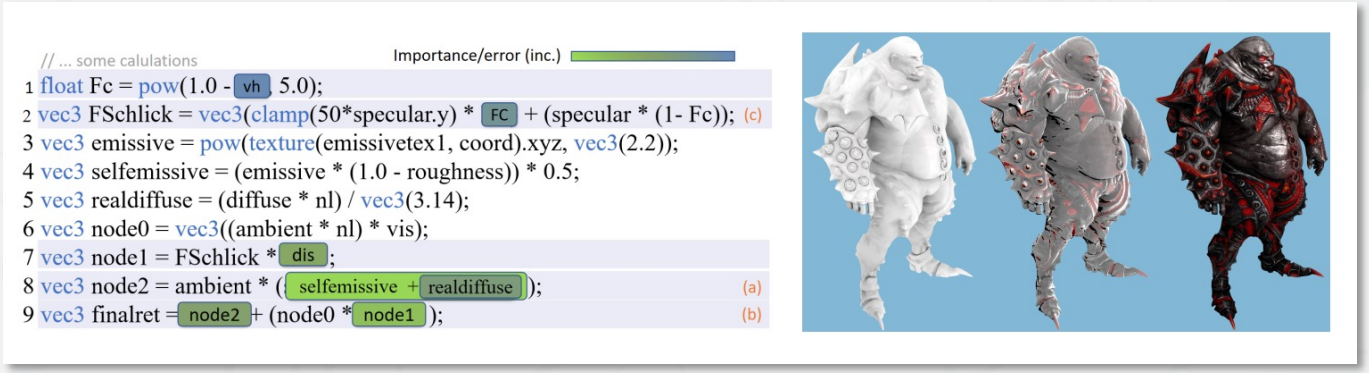


科研：支撑各类图形绘制算法

图形绘制

绘制流水线

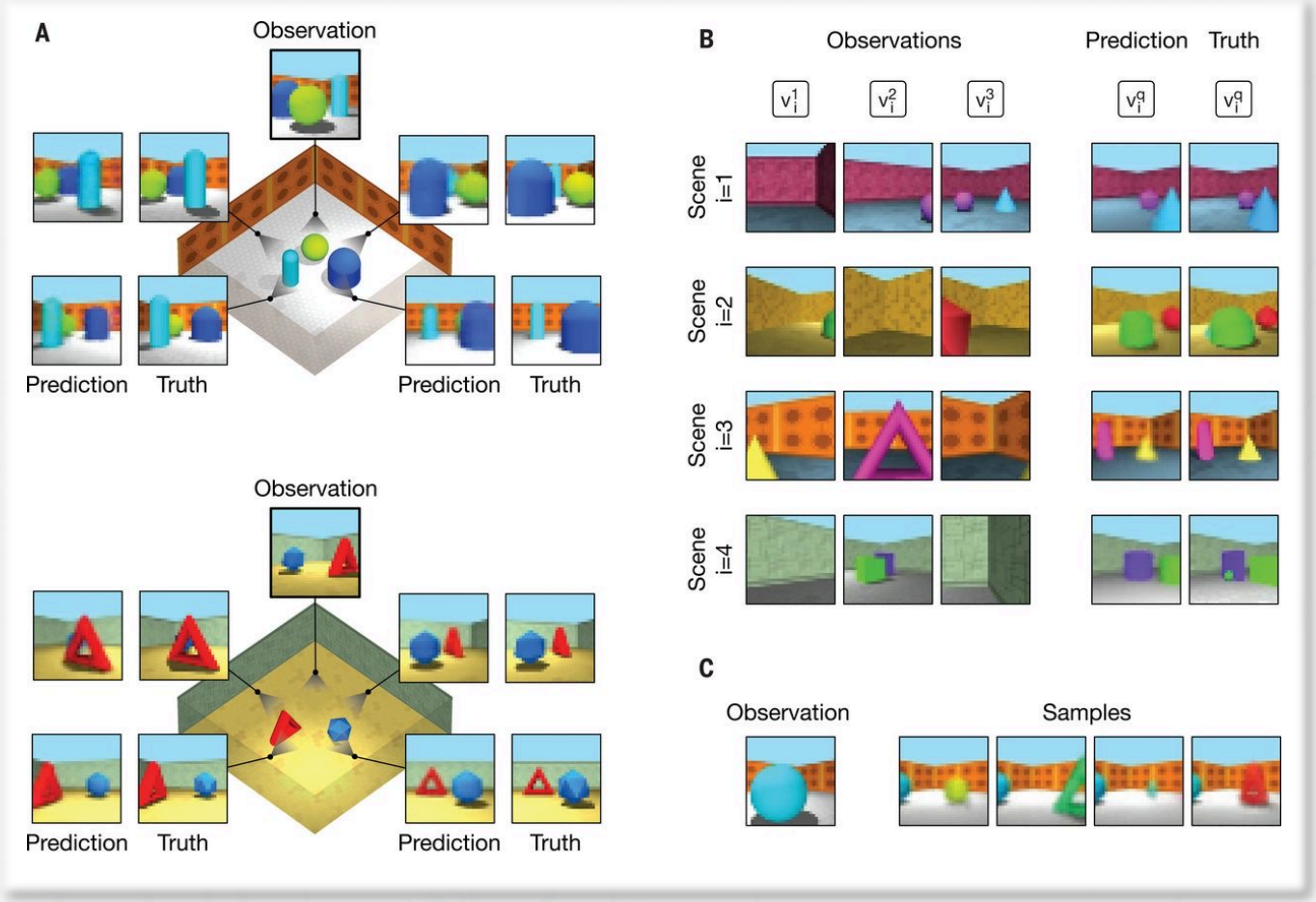




科研：以流水线作为研究对象

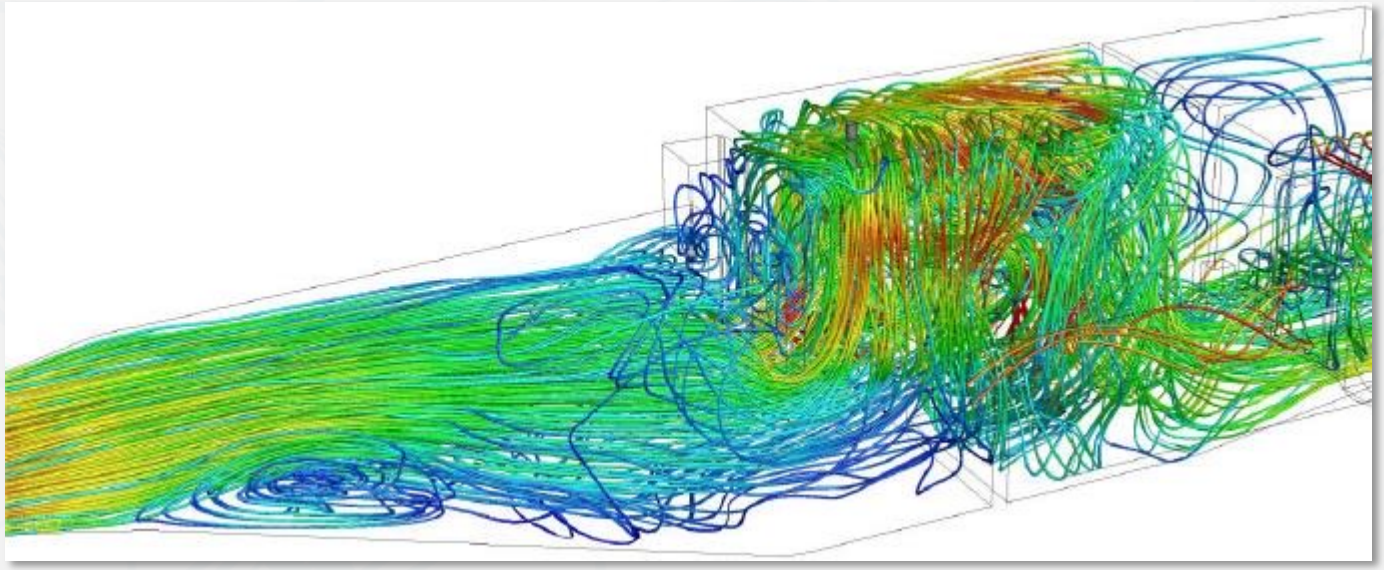
图形绘制

绘制流水线



科研：数据集的生成工具

绘制流水线



科研：科研结果的可视化工具

GAMES106讲什么？

工业

绘制流水线

学术

理论 + 实践

GAMES 106

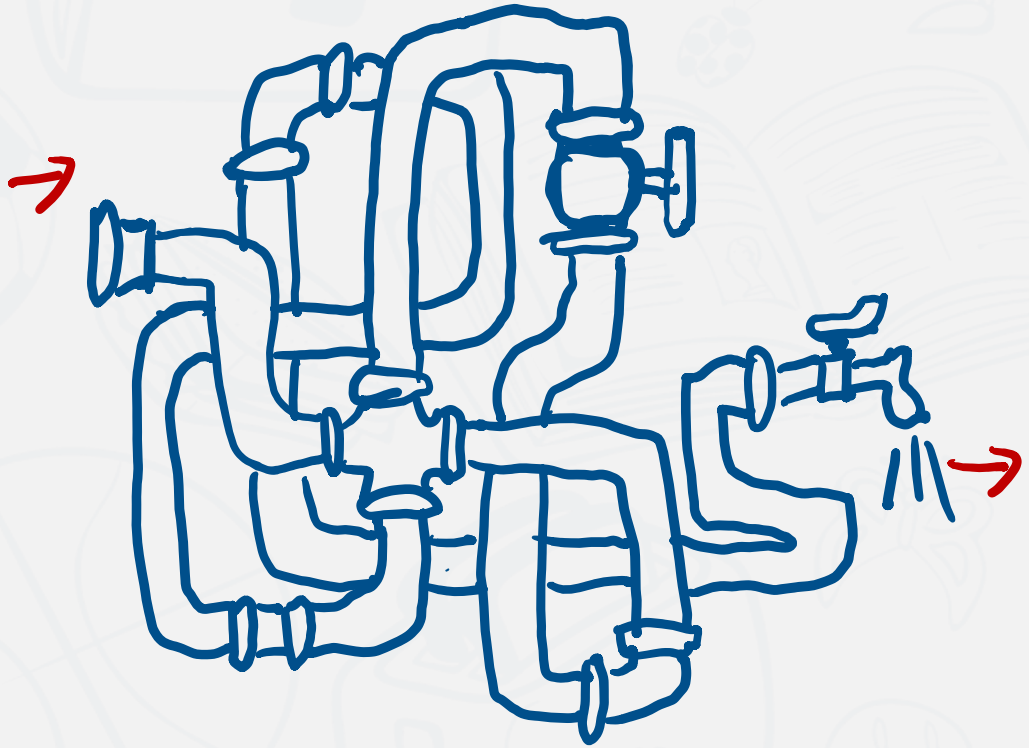
现代图形绘制流水线原理与实践

L1 2023/4/5

如何感性认识绘制流水线？

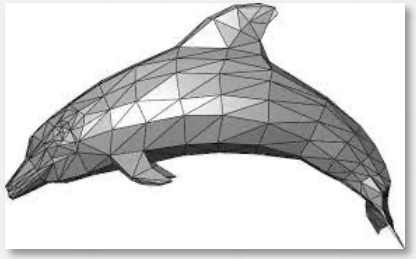
L1 2023/4/5

流水线



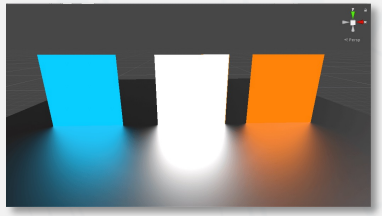
固定绘制流水线

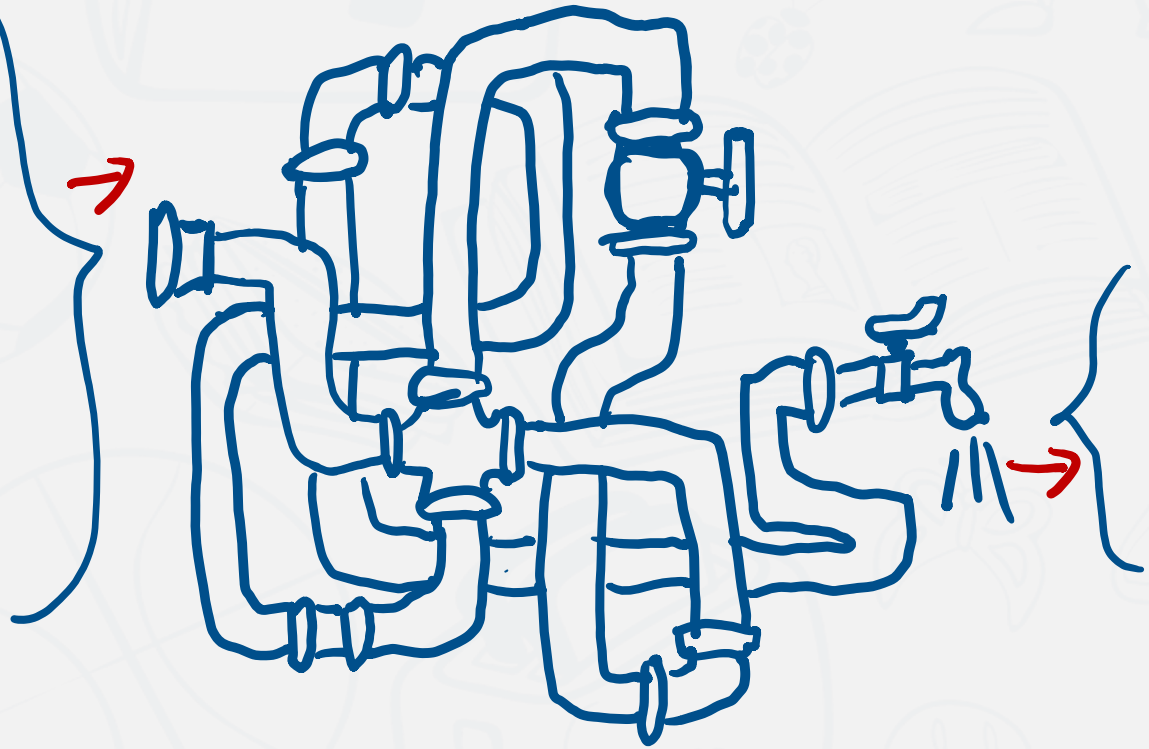
几何



纹理

光照



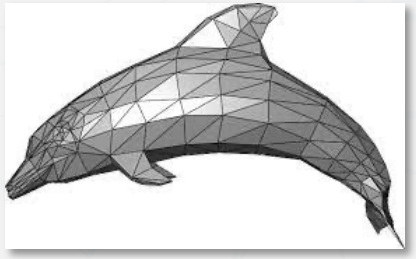


图片

图片=F(几何，纹理，光照）

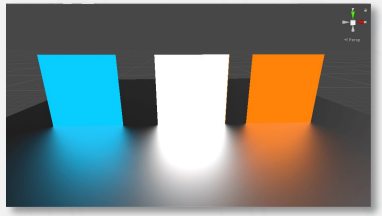
可编程绘制流水线

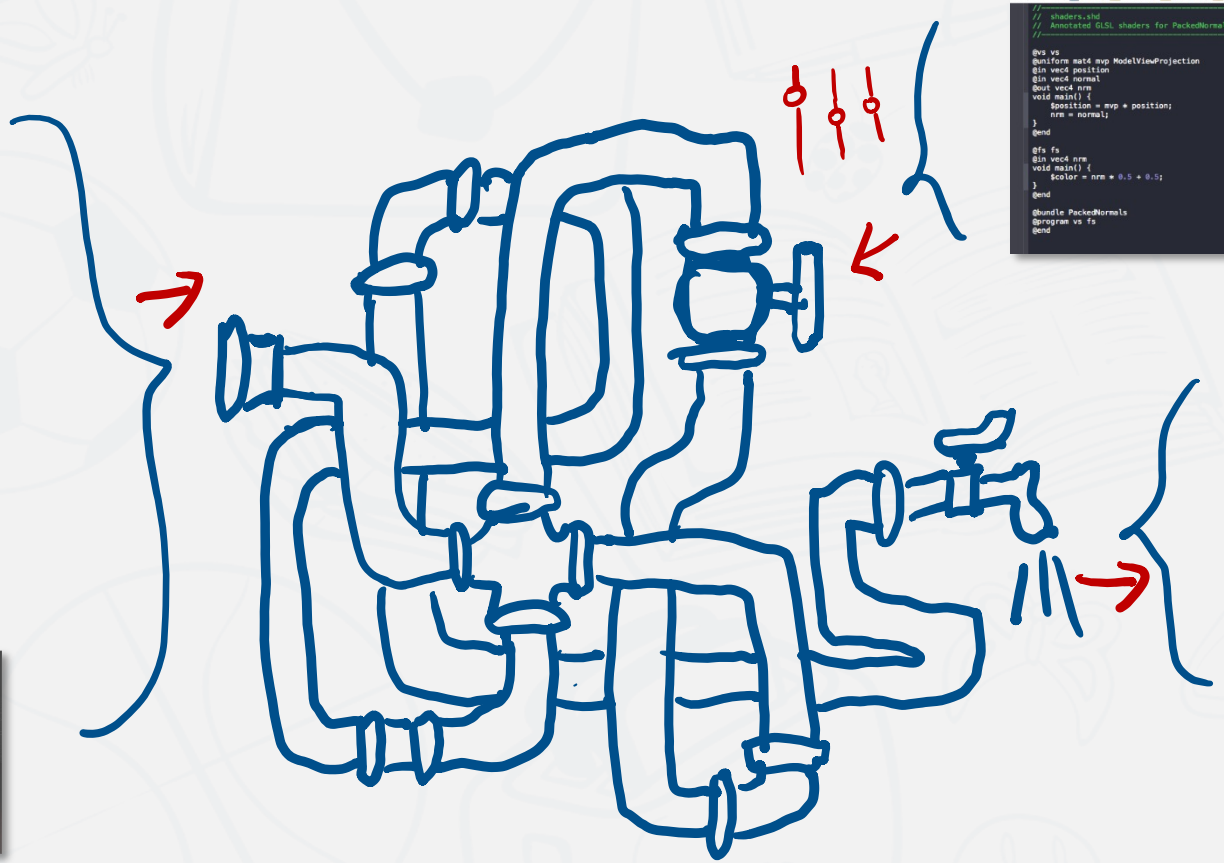
几何



纹理

光照





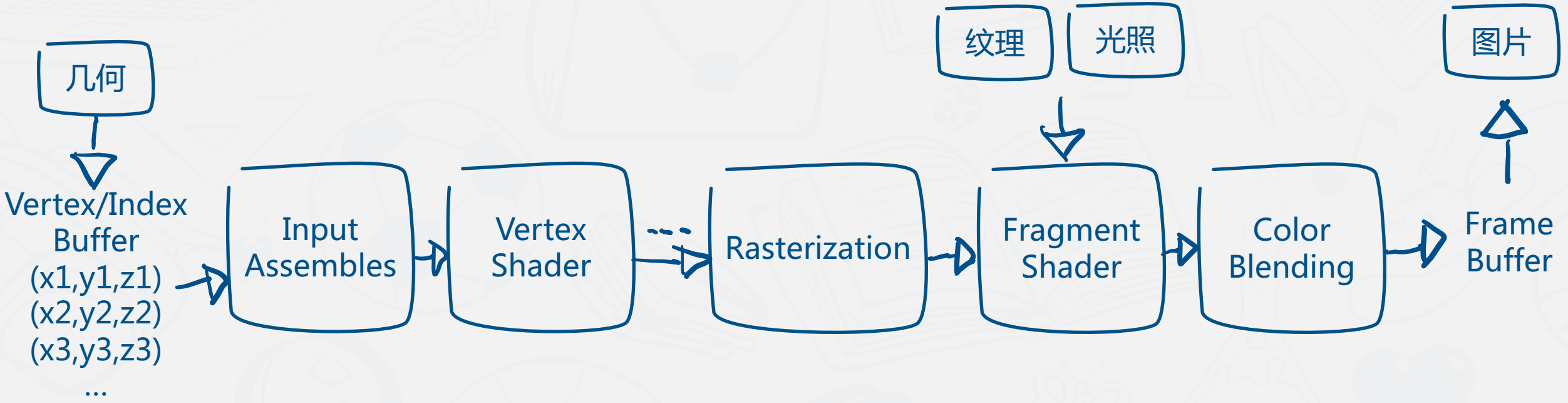
着色器

```
//
//  shaders.shd
//  Annotated GLSL shaders for PackedNormals sample
//
@vs vs
@uniform mat4 mvp ModelViewProjection
@in vec4 position
@in vec4 normal
@out vec4 nrm
void main() {
    $position = mvp * position;
    nrm = normal;
}
@end

@fs fs
@in vec4 nrm
void main() {
    $color = nrm * 0.5 + 0.5;
}
@end

@bundle PackedNormals
@program vs fs
@end
```

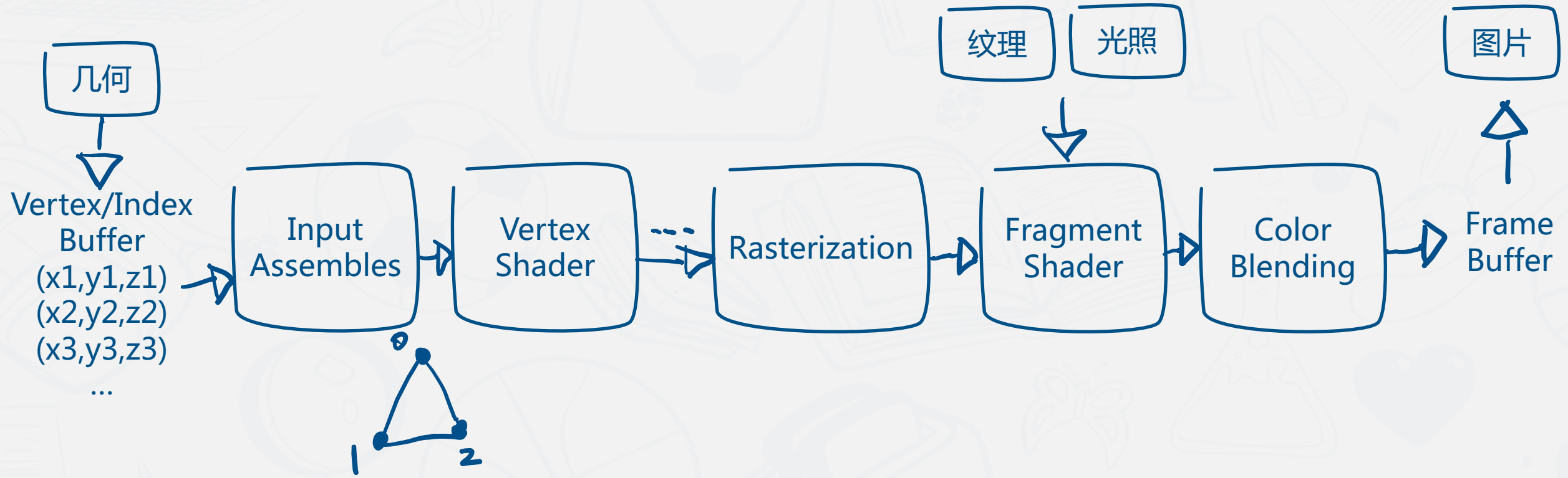
图片

图片=F(几何，纹理，光照；着色器)

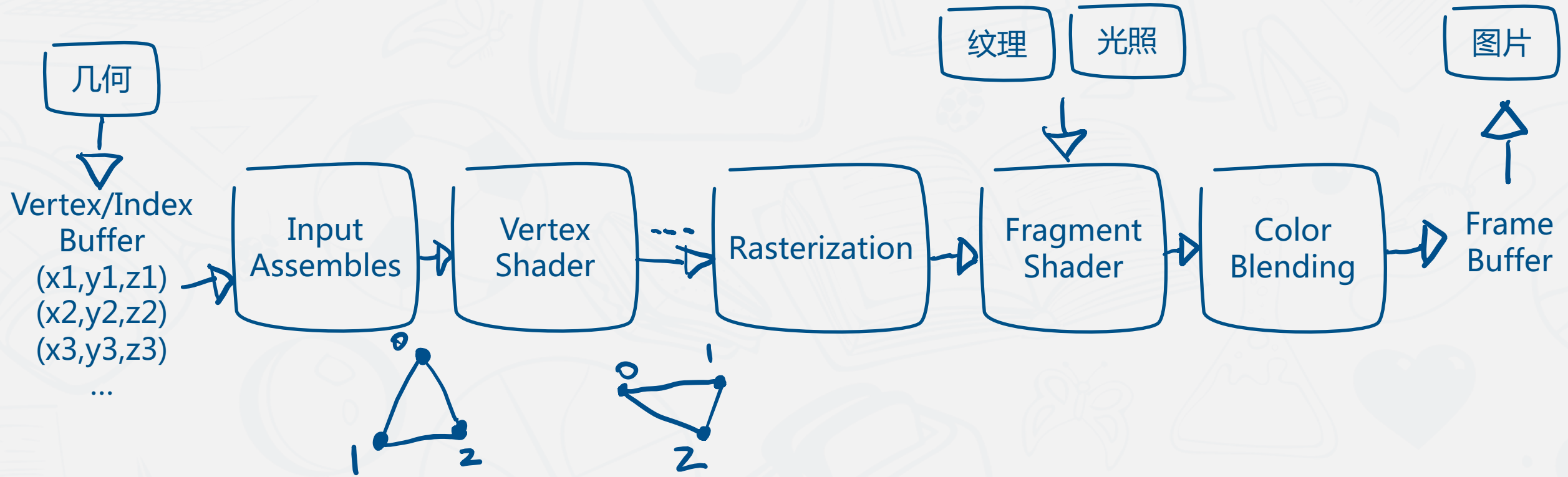
可编程绘制流水线


几何
Vertex/Index
Buffer
(x1,y1,z1)
(x2,y2,z2)
(x3,y3,z3)
...
Input
Assembles
Vertex
Shader
Rasterization
纹理
光照
Fragment
Shader
Color
Blending
Frame
Buffer
图片

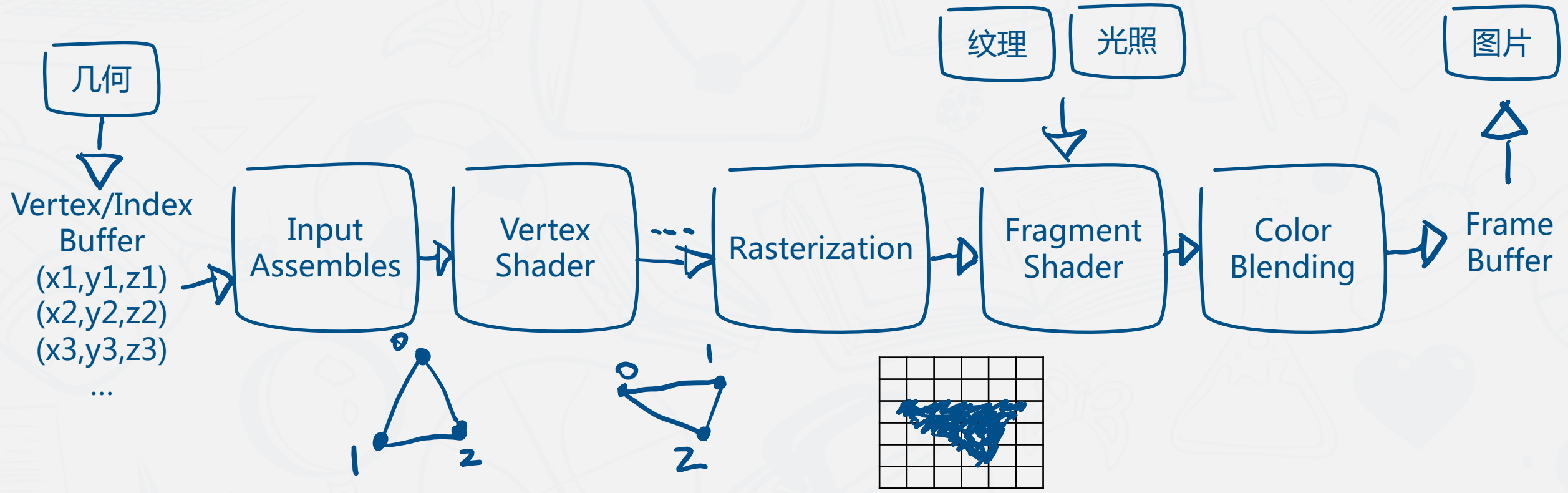
可编程绘制流水线


几何
Vertex/Index
Buffer
(x1,y1,z1)
(x2,y2,z2)
(x3,y3,z3)
...
Input
Assembles
Vertex
Shader
Rasterization
纹理
光照
Fragment
Shader
Color
Blending
Frame
Buffer
图片
0
1
2

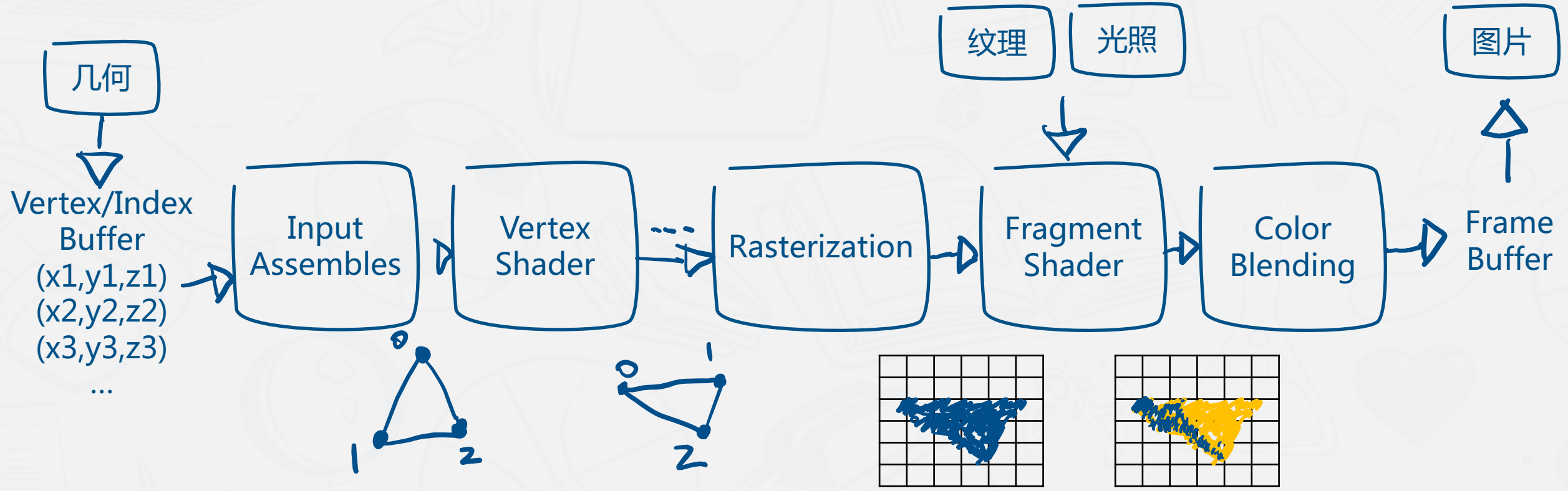
可编程绘制流水线


几何
Vertex/Index
Buffer
(x1,y1,z1)
(x2,y2,z2)
(x3,y3,z3)
...
Input
Assembles
Vertex
Shader
Rasterization
纹理
光照
Fragment
Shader
Color
Blending
图片
Frame
Buffer
0
1
2
0
1
2

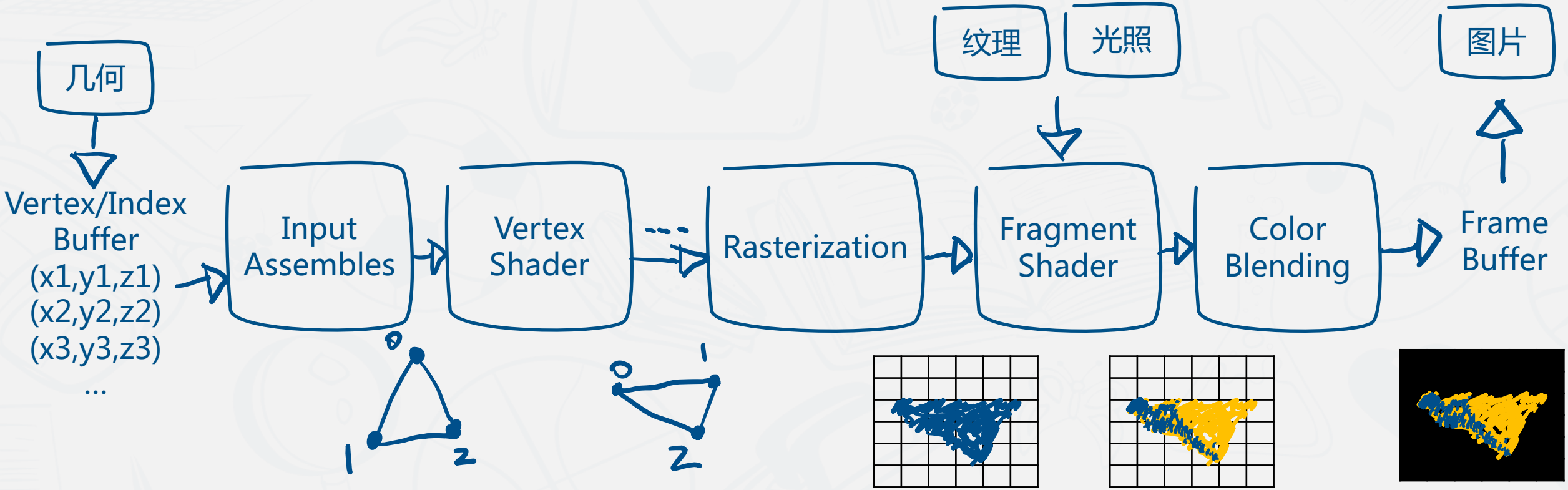
可编程绘制流水线


几何
Vertex/Index Buffer
(x1,y1,z1)
(x2,y2,z2)
(x3,y3,z3)
...
Input Assembles
Vertex Shader
Rasterization
纹理
光照
Fragment Shader
Color Blending
图片
Frame Buffer
0
1
2
0
1
2

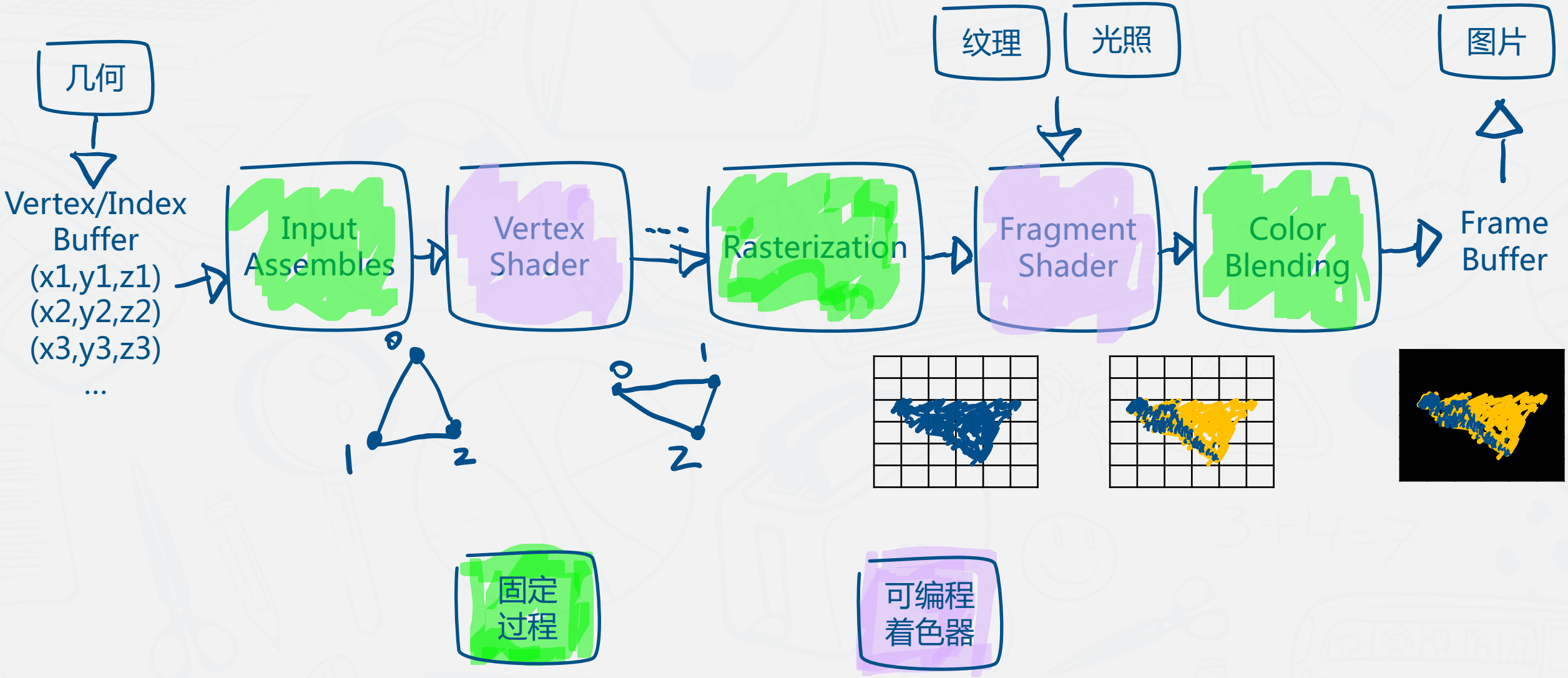
可编程绘制流水线


几何
Vertex/Index
Buffer
(x1,y1,z1)
(x2,y2,z2)
(x3,y3,z3)
...
Input
Assembles
Vertex
Shader
Rasterization
纹理
光照
Fragment
Shader
Color
Blending
图片
Frame
Buffer
0
1
2
0
1
2

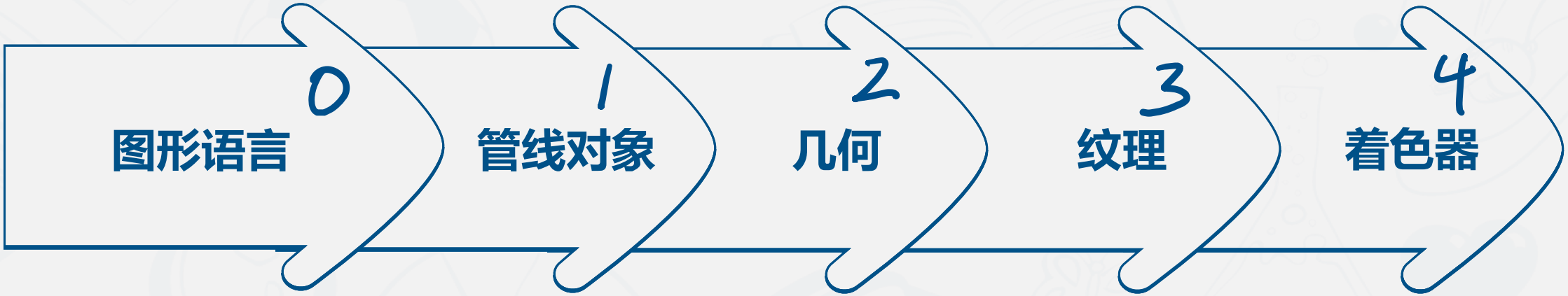
可编程绘制流水线


几何
Vertex/Index
Buffer
(x1,y1,z1)
(x2,y2,z2)
(x3,y3,z3)
...
Input
Assembles
Vertex
Shader
Rasterization
纹理
光照
Fragment
Shader
Color
Blending
图片
Frame
Buffer
0
1
2
0
1
2

可编程绘制流水线


几何
Vertex/Index Buffer
(x1,y1,z1)
(x2,y2,z2)
(x3,y3,z3)
...
Input Assembles
Vertex Shader
Rasterization
纹理
光照
Fragment Shader
Color Blending
图片
Frame Buffer
0
1
2
固定过程
可编程着色器

课程分解



高涛　　袁亚振　　高希峰　　胡义伟　　霍宇驰

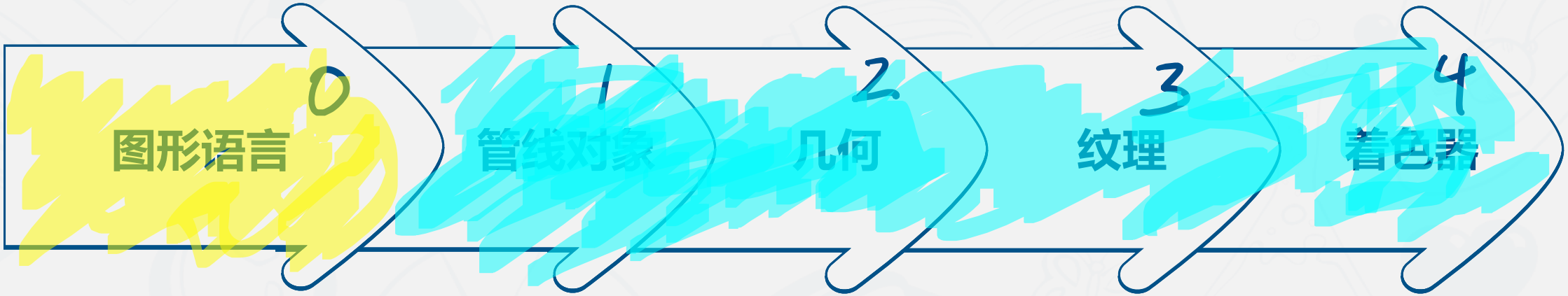
图片=F(几何，纹理，光照；着色器）

1 2 3 4

课程分解

加法

减法



高涛　　袁亚振　　高希峰　　胡义伟　　霍宇驰

图片=F(几何，纹理，光照；着色器）

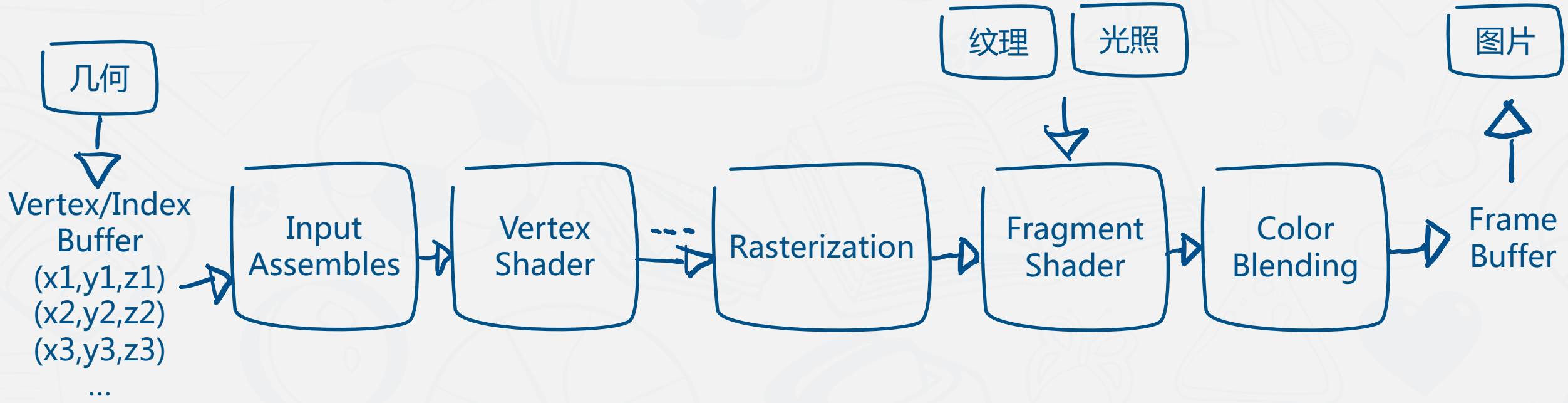
1 2 3 4

课程分解

构建

优化

如何感性认识绘制流水线？

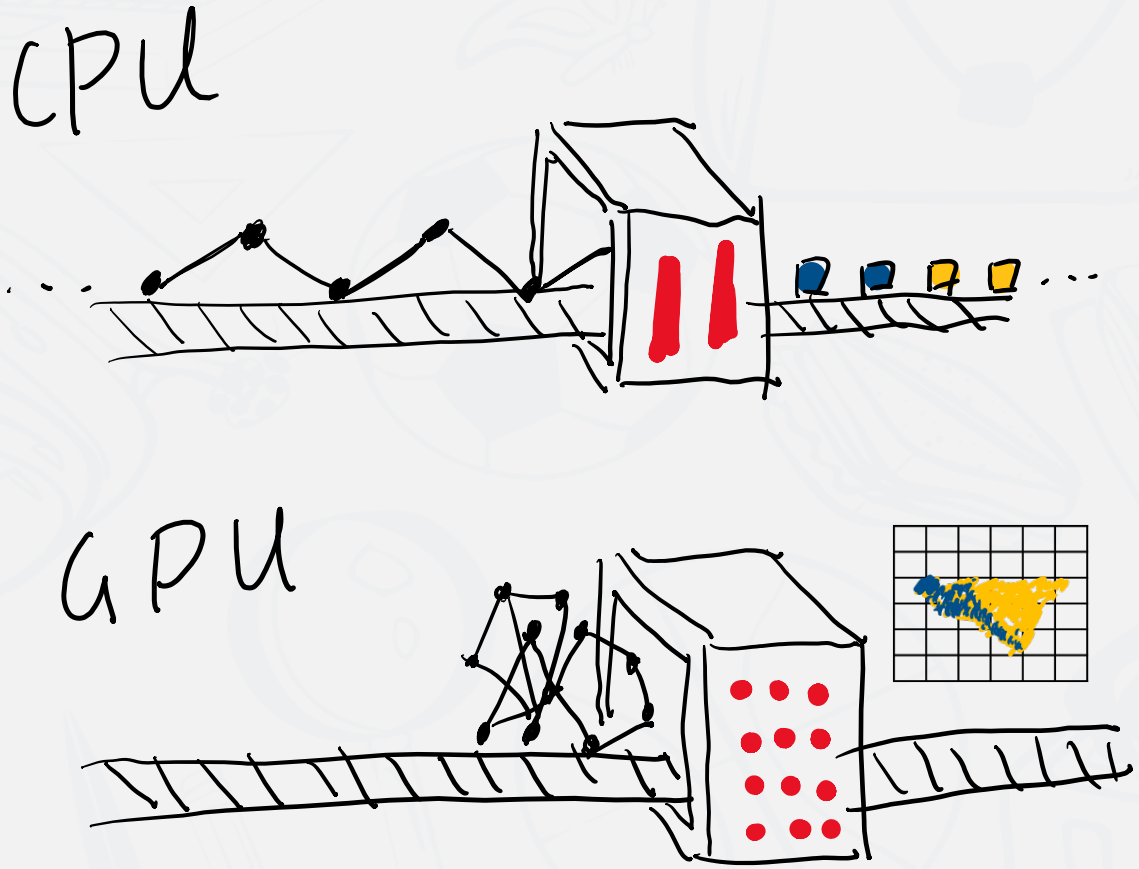


图片=F(几何，纹理，光照；着色器）

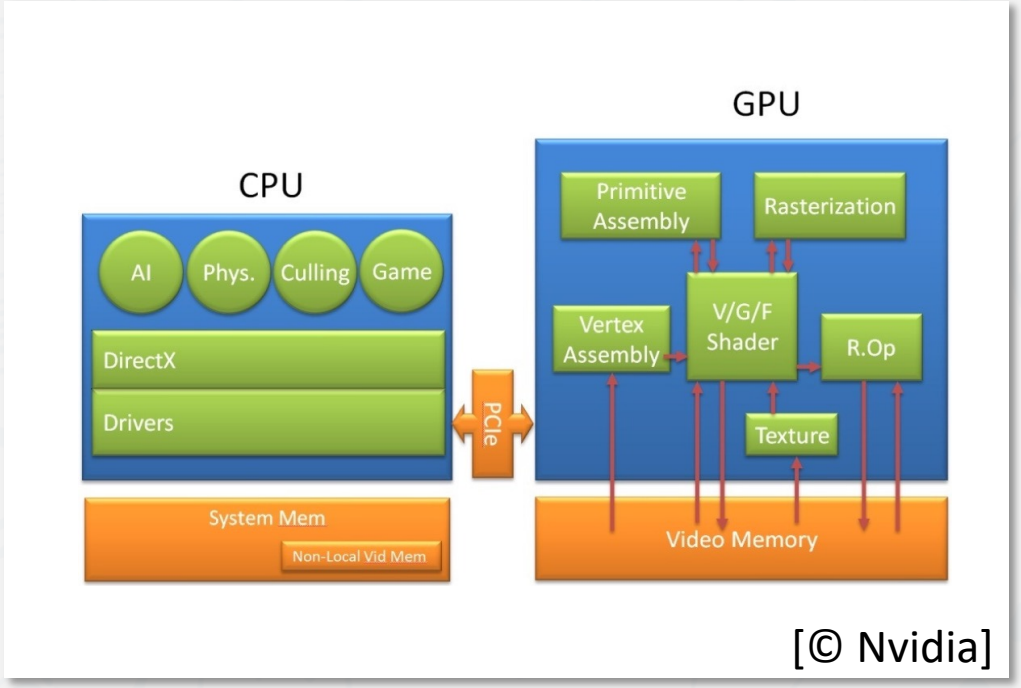
绘制流水线是如何演变的？

L1 2023/4/5

经典绘制流水线



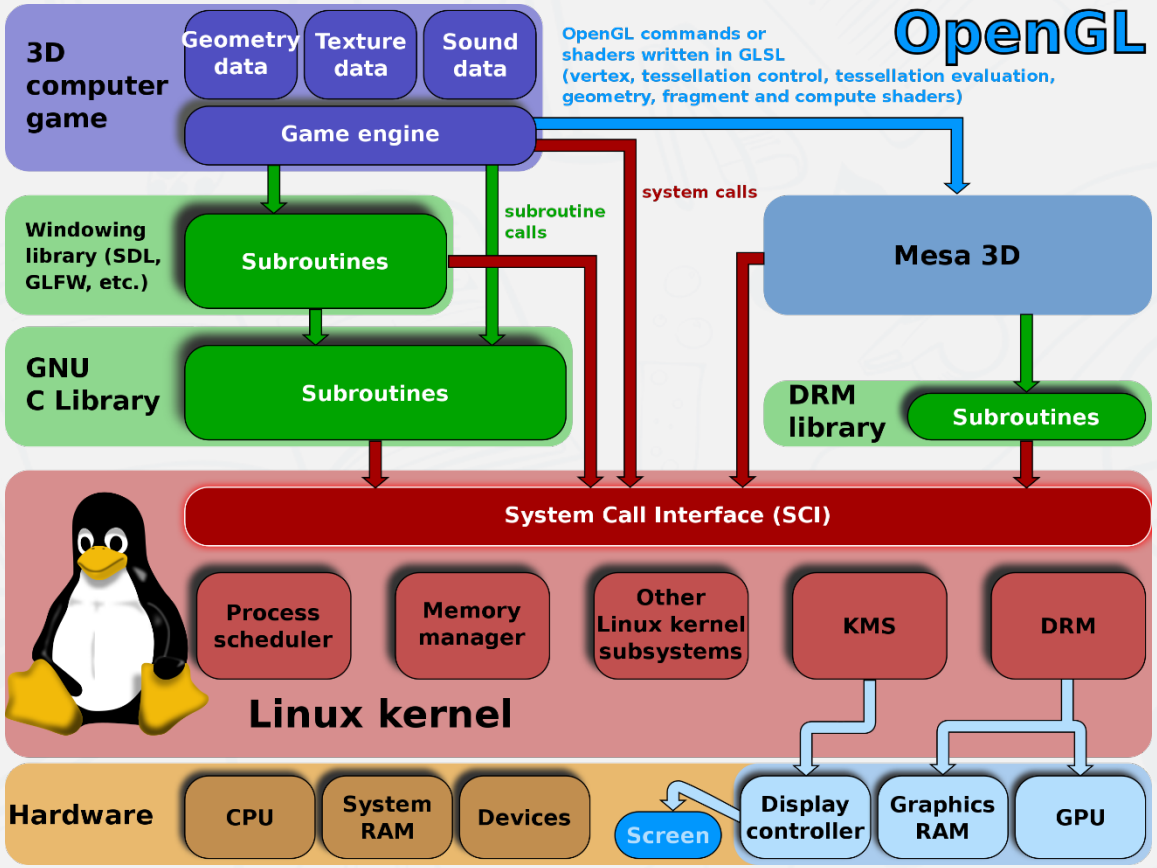
绘制流水线的演变



经典绘制流水线

绘制流水线的演变

I. **固定流水线、顶点数组、显示列表、纹理对象、帧缓冲区…**
   - **OpenGL 1.0 (1992)**



OpenGL

经典绘制流水线

绘制流水线的演变

I. **固定流水线、顶点数组、显示列表、纹理对象、帧缓冲区…**
   - **OpenGL 1.0 (1992)**
   - **3dfx Glide 1.0 (1995)**
   - **DirectX 1.0 (1995)**

GLQuake (1997)

经典绘制流水线

绘制流水线的演变

I. 固定流水线、顶点数组、显示列表、纹理对象、帧缓冲区…
   - OpenGL 1.0 (1992)
   - 3dfx Glide 1.0 (1995)
   - DirectX 1.0 (1995)

II. 着色语言、像素着色器
   - OpenGL 2.0, GLSL (2004)
   - DirectX 9.0, HLGL (2002)

World of Warcraft (2004)

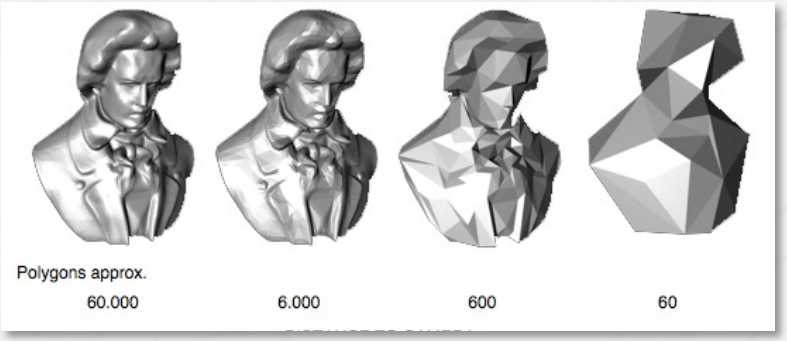
经典绘制流水线

绘制流水线的演变

I. 固定流水线、顶点数组、显示列表、纹理对象、帧缓冲区…
   - OpenGL 1.0 (1992)
   - 3dfx Glide 1.0 (1995)
   - DirectX 1.0 (1995)

II. 着色语言、像素着色器
   - OpenGL 2.0, GLSL (2004)
   - DirectX 9.0, HLGL (2002)

III. 几何着色器
   - OpenGL 3.0 (2008)
   - DirectX 10 (2006)



Crysis 2 (2011)

经典绘制流水线

绘制流水线的演变

I. 固定流水线、顶点数组、显示列表、纹理对象、帧缓冲区…
   - OpenGL 1.0 (1992)
   - 3dfx Glide 1.0 (1995)
   - DirectX 1.0 (1995)

II. 着色语言、像素着色器
   - OpenGL 2.0, GLSL (2004)
   - DirectX 9.0, HLGL (2002)

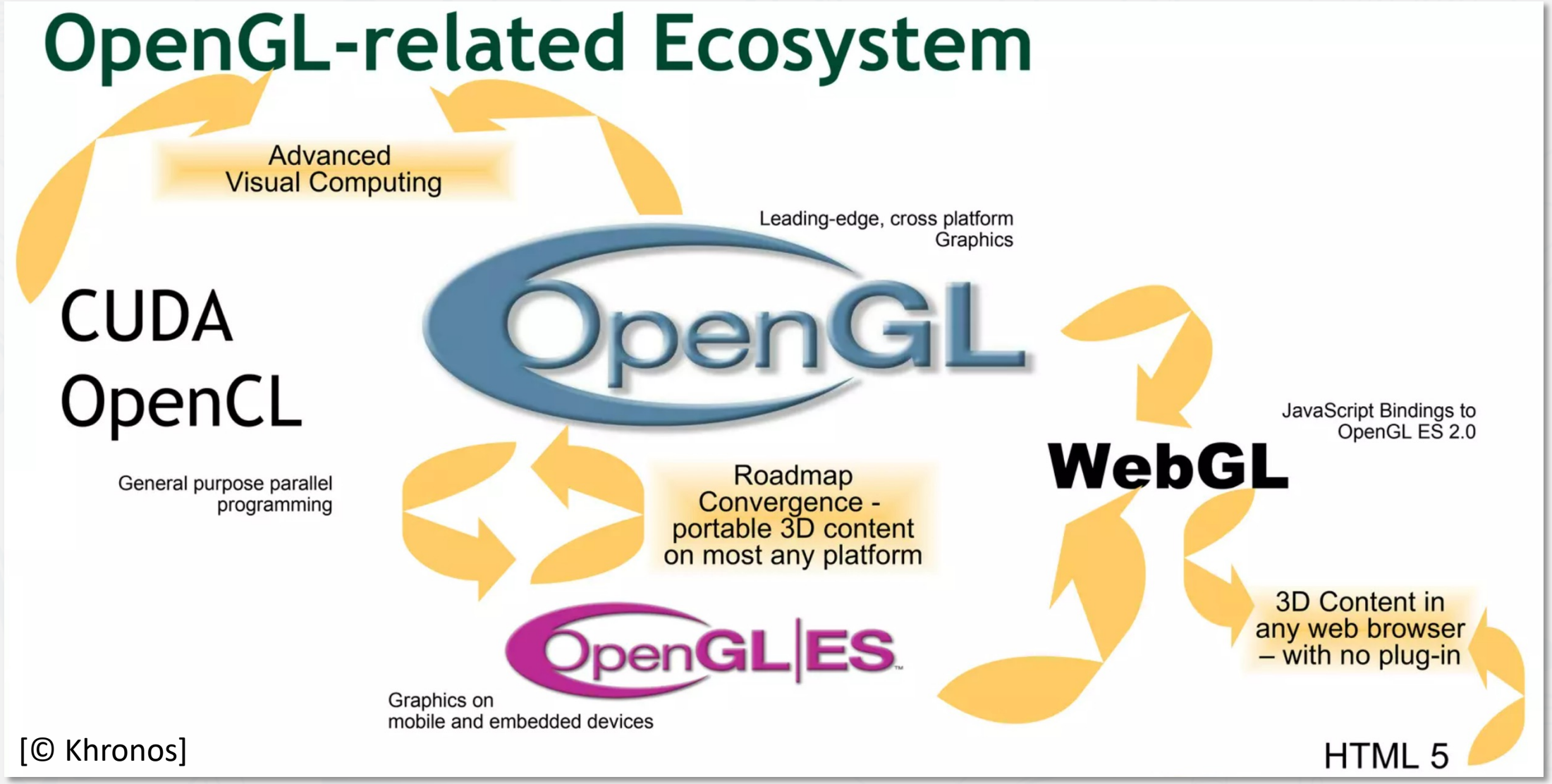
III. 几何着色器
   - OpenGL 3.0 (2008)
   - DirectX 10 (2006)

IV. 曲面细分
   - OpenGL 4.0 (2010)
   - DirectX 11 (2009)

The Witcher 3 (2015)

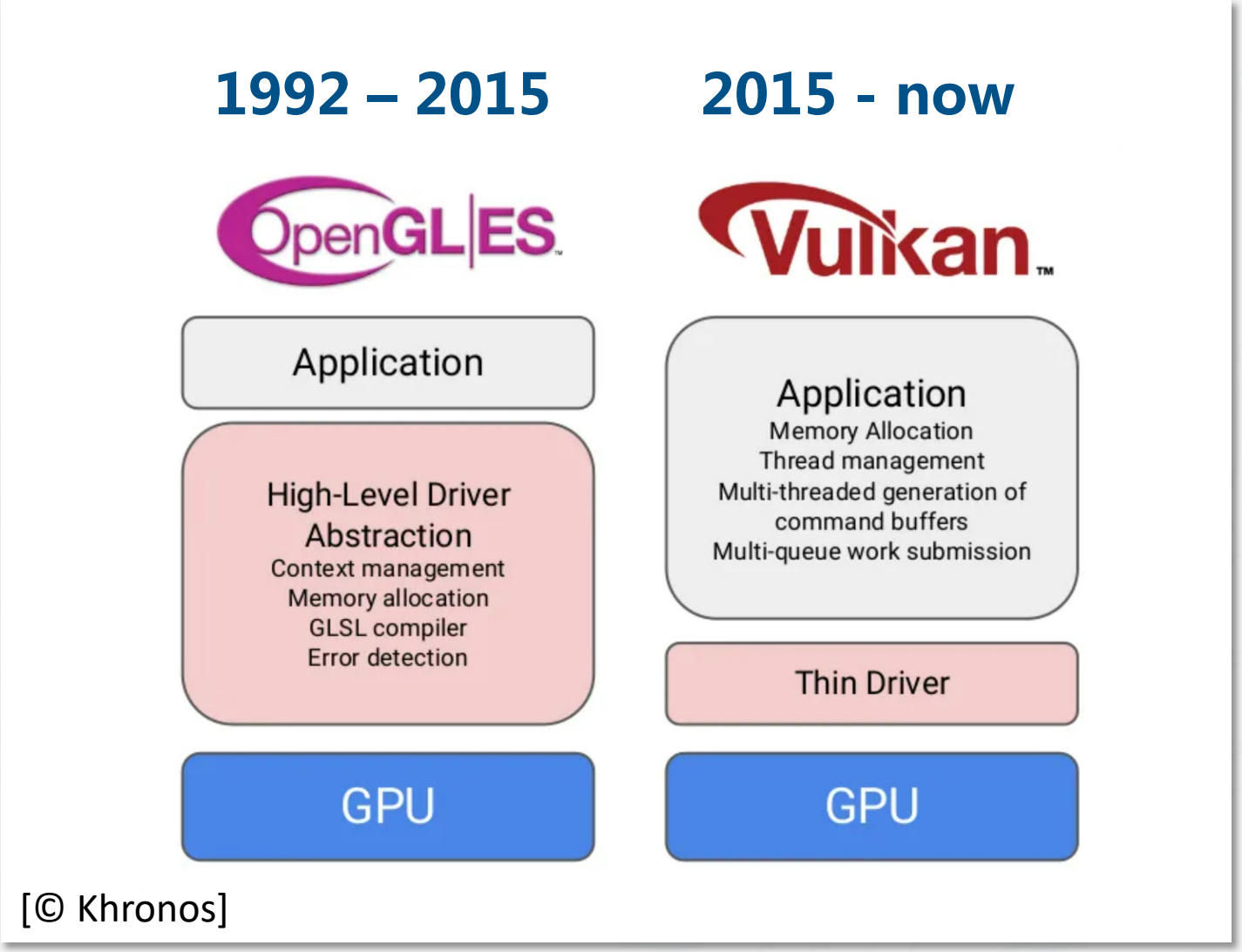
经典绘制流水线

绘制流水线的演变



现代绘制流水线

绘制流水线的演变



[© Khronos]

现代绘制流水线

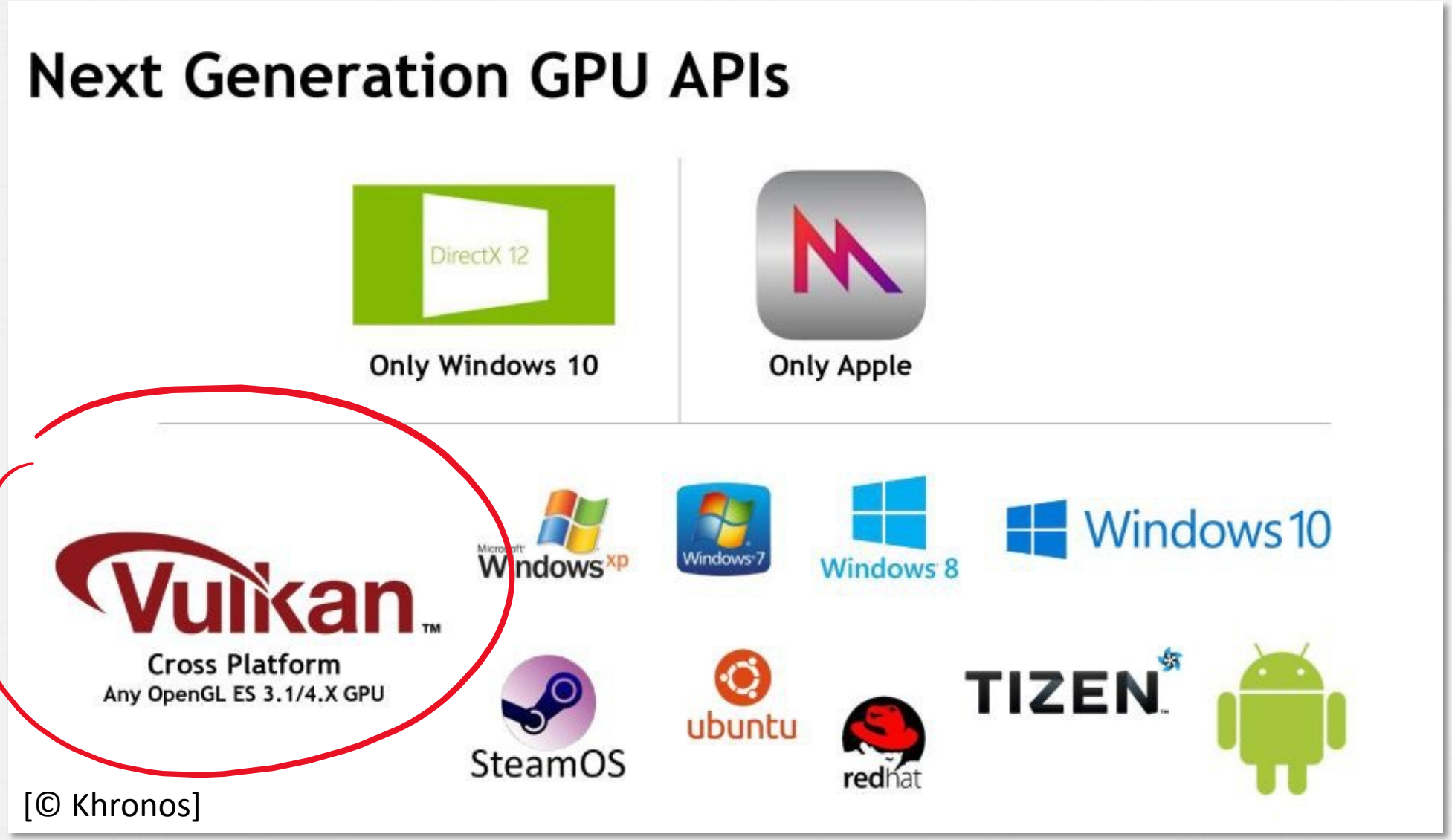
绘制流水线的演变

	
面向图形绘制架构，基于分离内存。	跨平台兼容手机等现代架构，支持统一内存。
隐式API：状态机模式，驱动负责状态控制、依赖追踪，错误检查等。性能不可控，通常较低。	显示API：程序员直接控制GPU的线程、内存、绘制队列。性能优化潜力更高。
单线程，不支持并行控制指令。	多线程，通过指令缓冲支持并行执行指令。
OpenGL兼容历史版本。	Vulkan放弃兼容历史版本。
硬件驱动提供本地GLSL/HLSL语言支持。	以中间代码作为编译目标，降低对驱动层的依赖，简化前端语言。
开发者需要考虑兼容不同硬件厂商的实现细节差异。	更简单的API，更鲁棒的跨平台兼容。

现代绘制流水线

绘制流水线的演变

[© PowerVR]

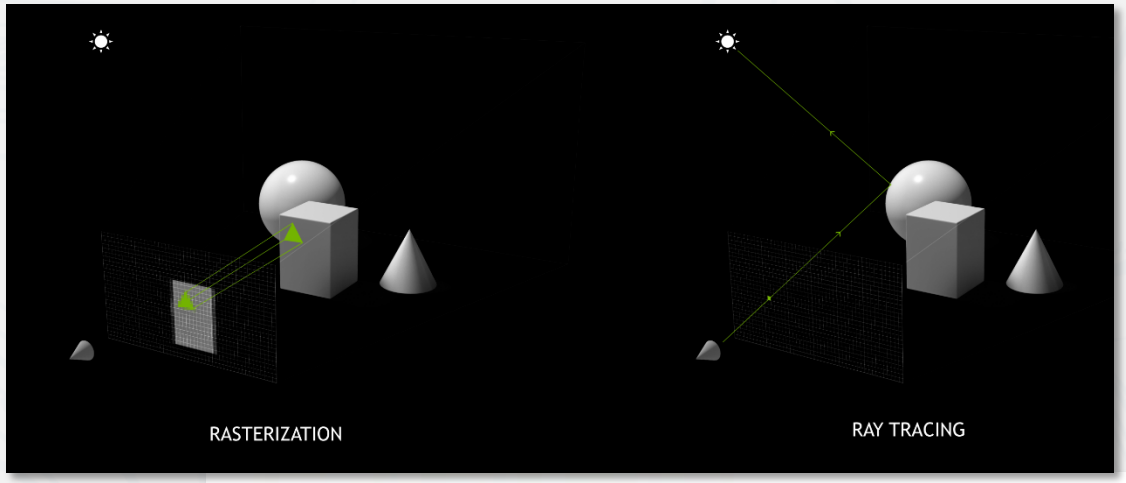


[© Khronos]

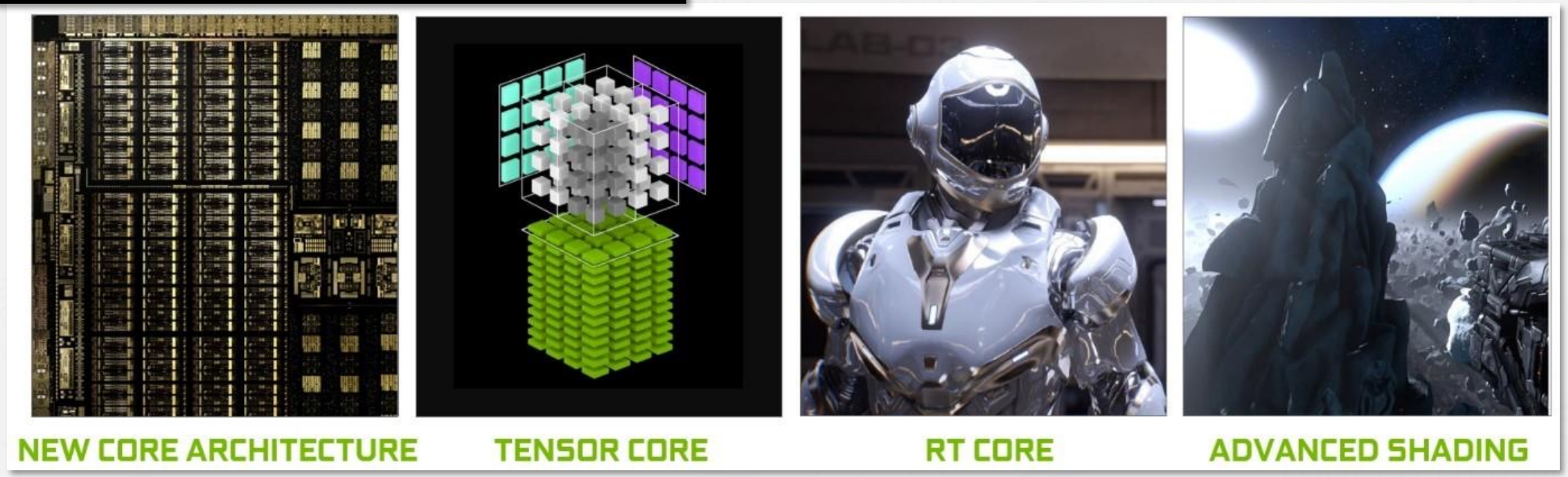
GAMES 106

混合绘制流水线

绘制流水线的演变



[© Nvidia]

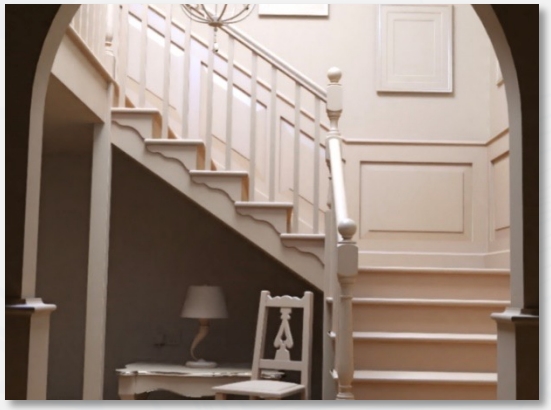


AI+绘制

绘制流水线的演变

插帧 [© Nvidia]

采样 [© ZJU]

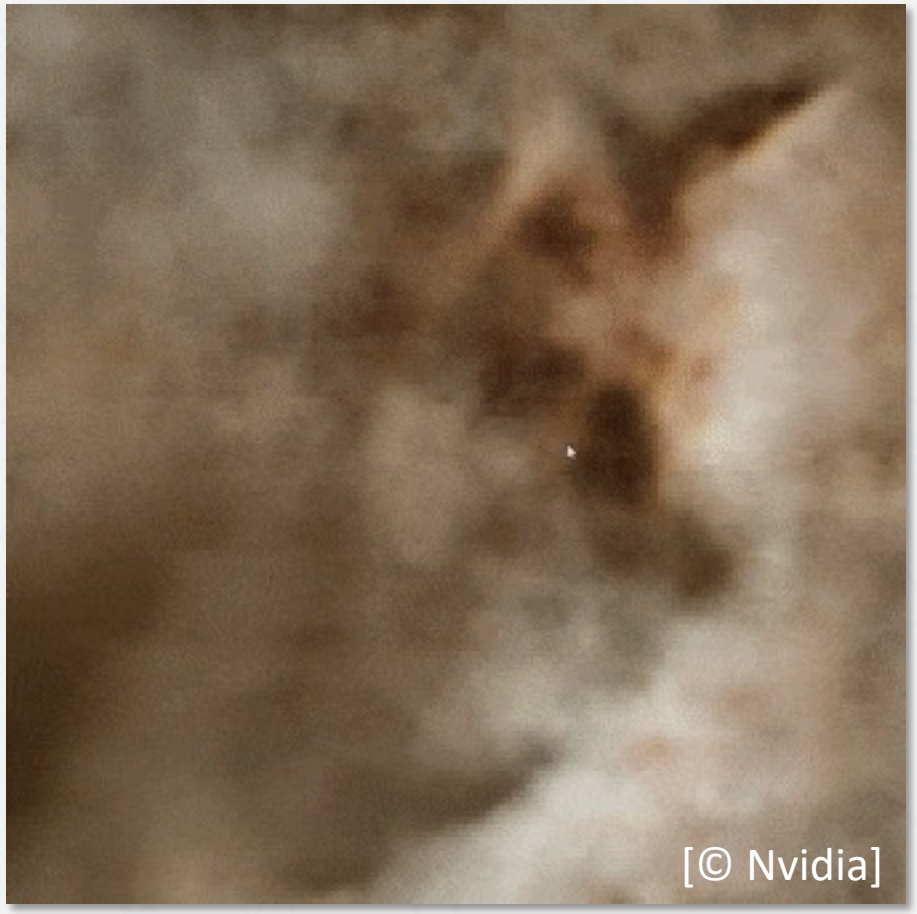


超分 [© Meta]

降噪 [© Disney]

隐式神经场表达

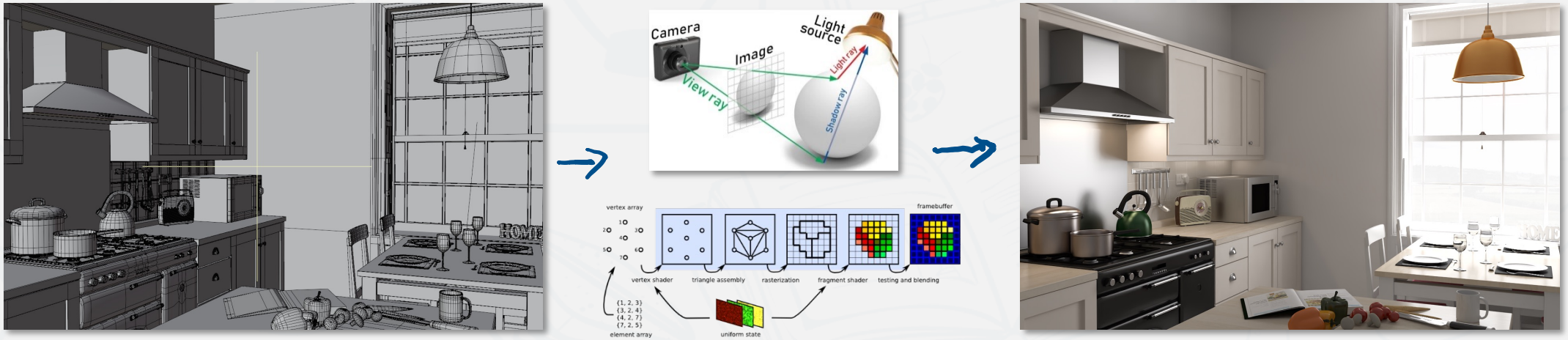
绘制流水线的演变



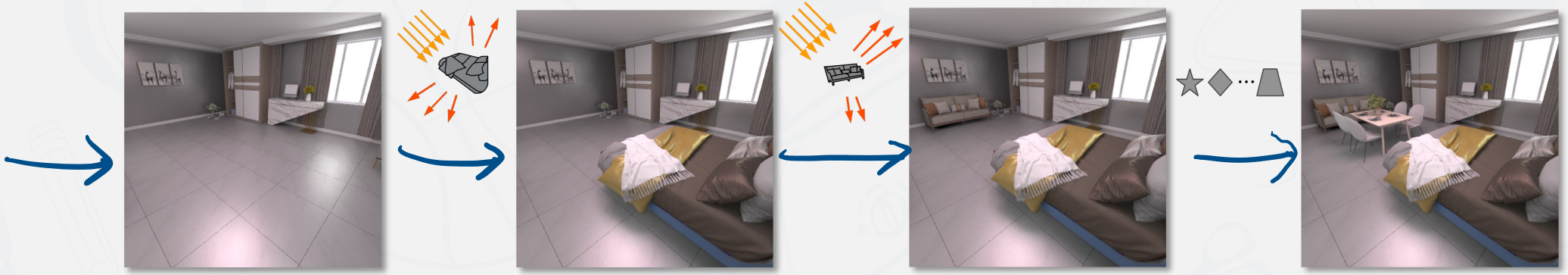
神经全局光照

绘制流水线的演变

传统绘制

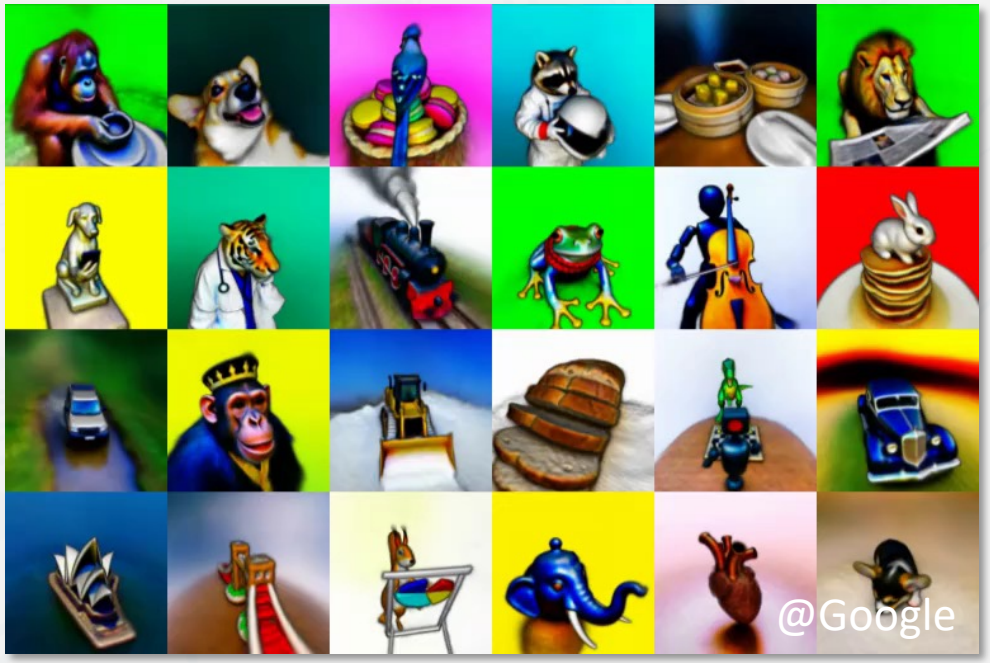


神经绘制



绘制流水线的演变

Text to Image



Text to 3D

绘制流水线是如何演变的？

固定流水线 **1990s** ⇒ 可编程流水线 **2000s** ⇒ 几何着色器 **2010s** ⇒ 现代流水线 **2020s**

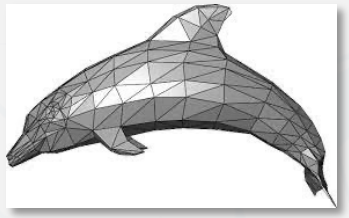
⇒ **Future**：神经绘制、混合绘制、隐式场、AIGC、深度绘制 ？

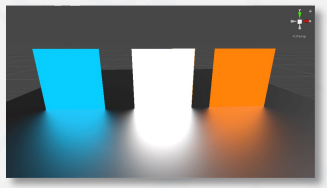
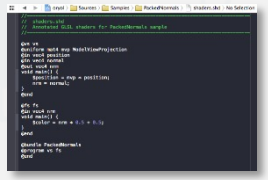
来一个作业

L1 2023/4/5

作业结构

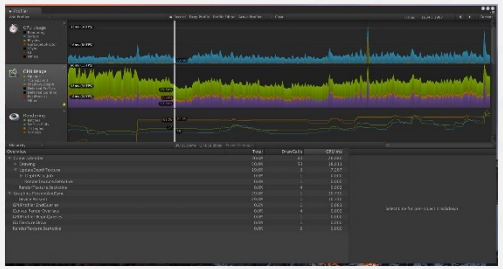
https://github.com/dodoleon/games106



① API · 创造

② 分析



③ 优化

④ …

管线对象·几何·纹理·着色器·

HOMEWORK 0

https://github.com/dodoleon/games106

- **画一个三角形**

HOMEWORK 0

https://github.com/dodoleon/games106

- **画一个三角形**

That's all !

VULKAN

谢谢您的欣赏

L1 2023/4/5