

MICROOH 麦可网

Android-从程序员到架构师之路

出品人：Sundy

讲师：高焕堂（台湾）

<http://www.microoh.com>

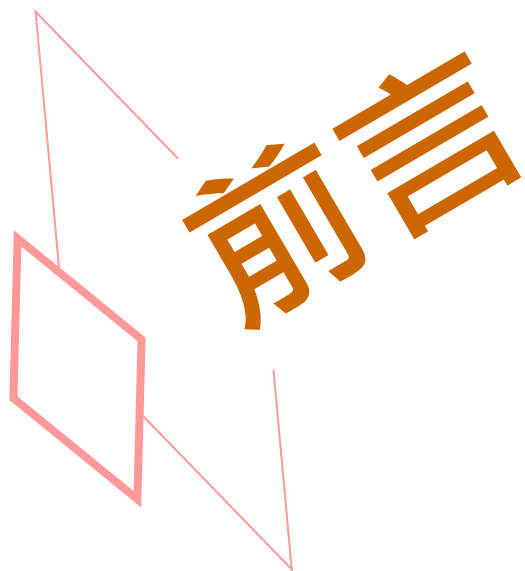
F03_b

观摩：SurfaceView小框架 的特殊性设计(b)

By 高煥堂

2、使用特殊性接口： Renderer

OpenGL ES C引擎

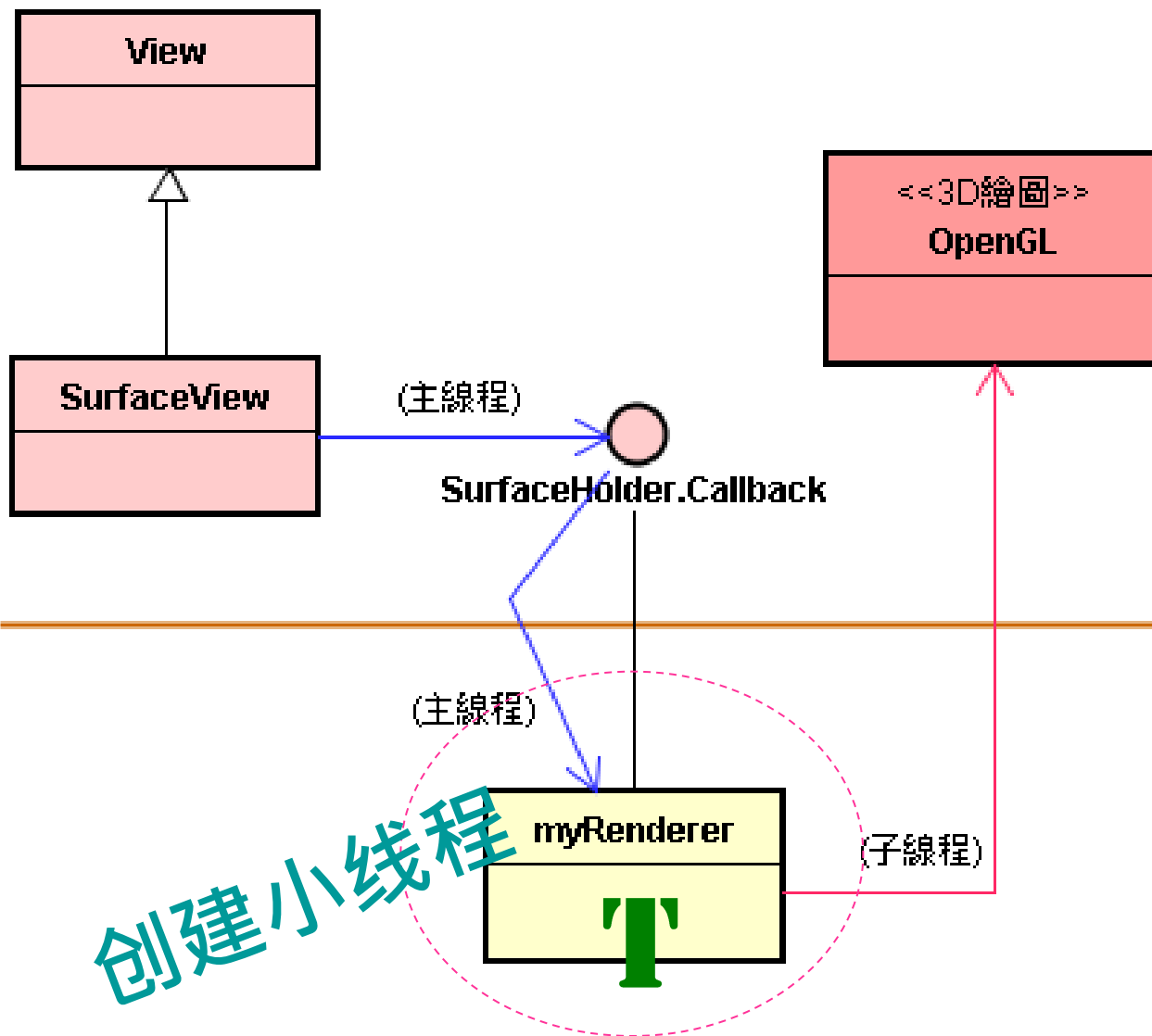


通用性接口不能顾及
特殊性的需求

- 例如，OpenGL的绘图是费时的工作，不适合使用UI线程(主线程)，于是产生了特殊需求：需要产生一个新线程(Thread)来担任绘图任务。此线程会去执行一个while循环，将不断地调整旋转角度，并呼叫MyCube对象，重新画出立方体的表面。

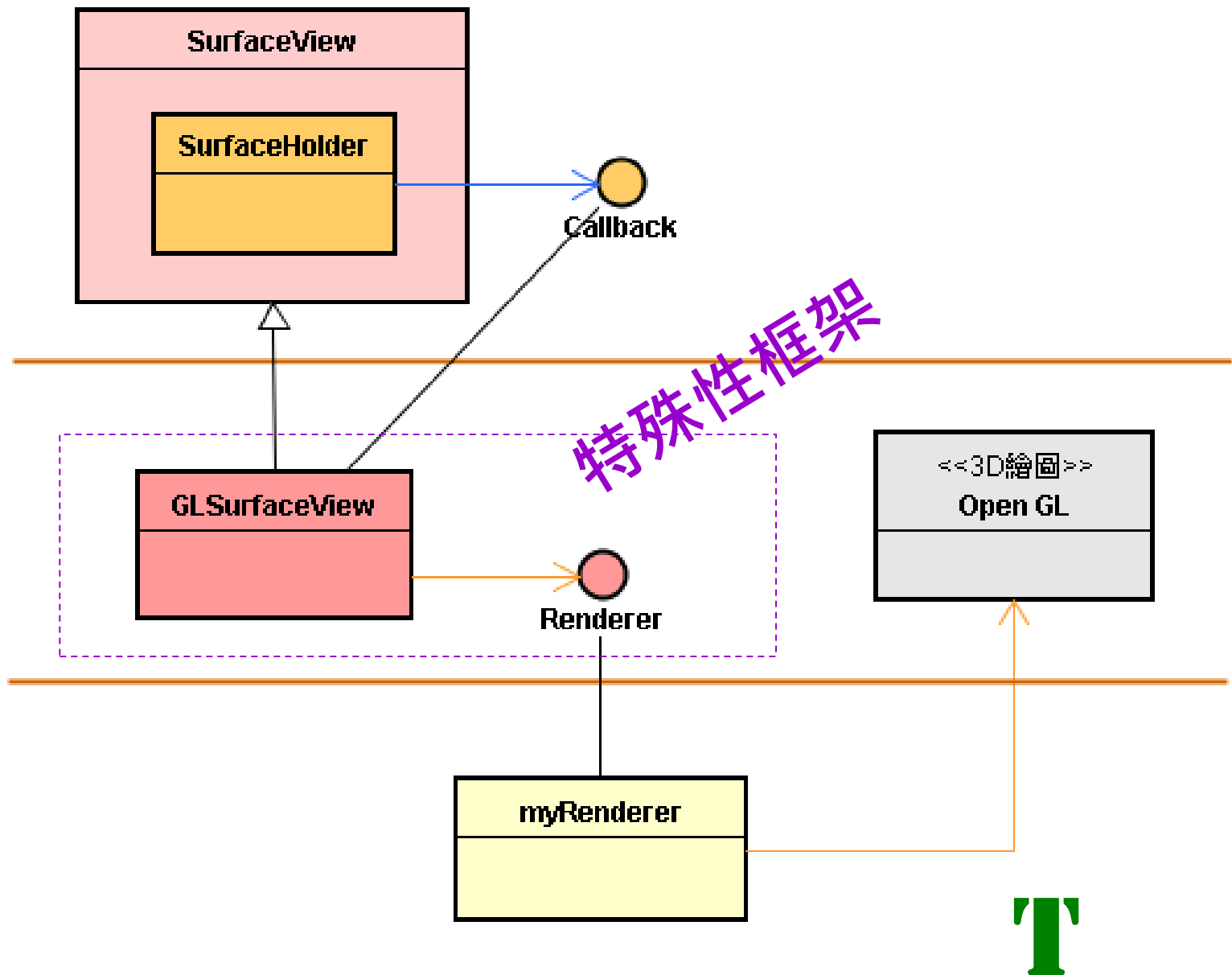
特殊性：
OpenGL需要小线程

- 这些特殊性就表达在App子类(就是<T>)里；于是，<T>就来创建一个小线程去执行OpenGL引擎的绘图任务。

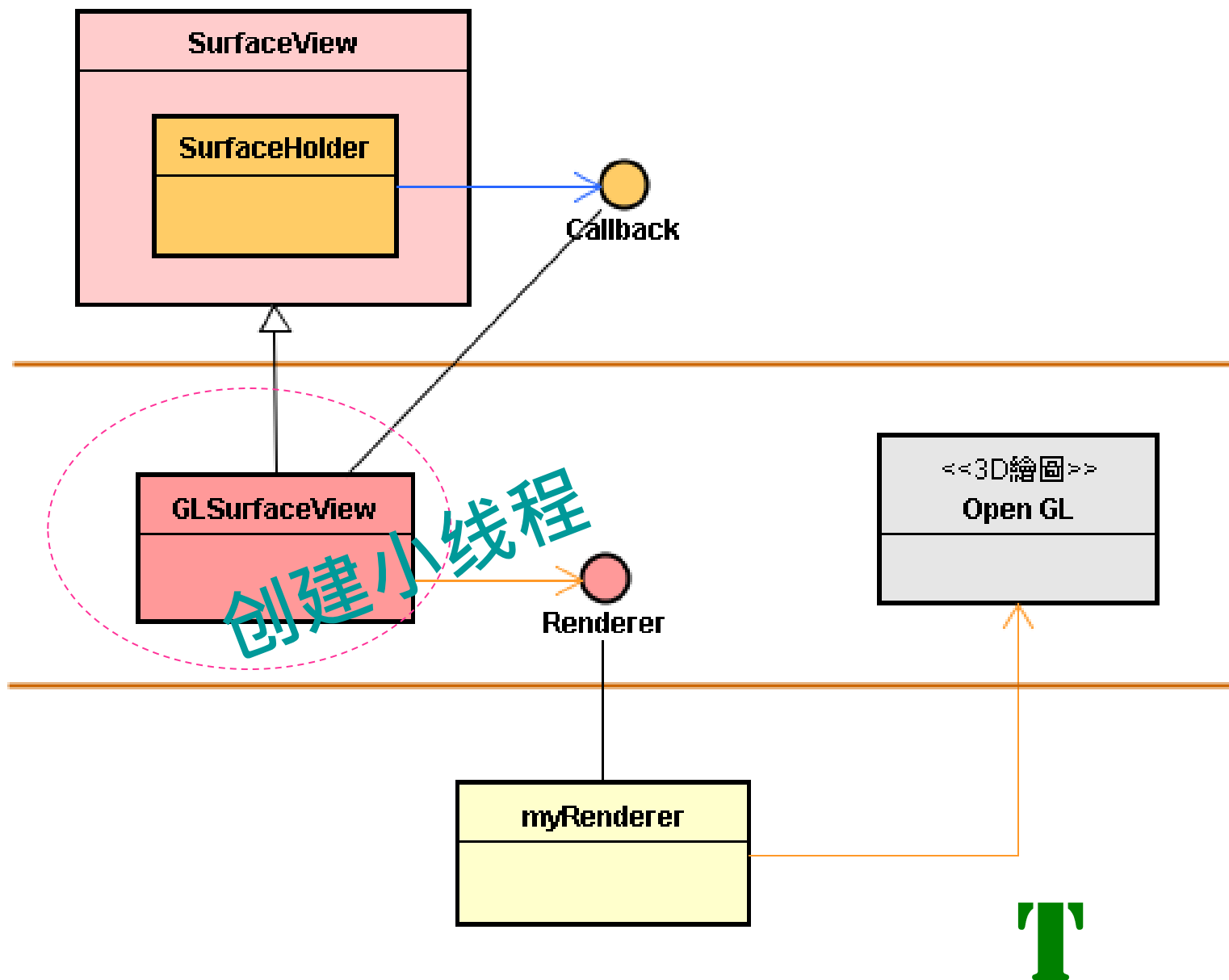


创建小线程

建立特殊性小框架，
兼容未来性与特殊性。



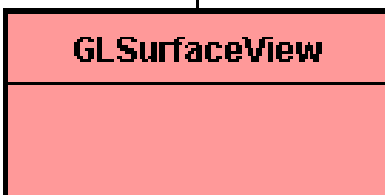
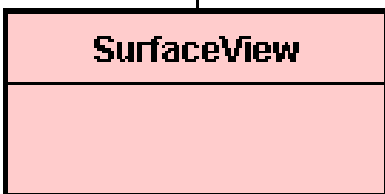
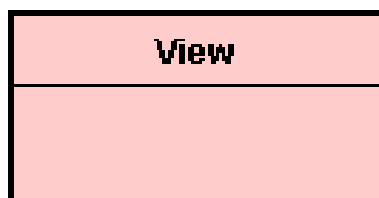
- 特殊性框架可以容纳特殊性的代码，因此可以在 `GLSurfaceView` 里创建小线程，担任执行OpenGL绘图的任务。



- 除了创建小线程之外，还可以将原来摆在<T>的代码，移到GLSurfaceView类(即Stub类)里。

GLSurfaceView和Renderer接口

- Java层的OpenGL ES版本速度比较慢。
- 想提高其3D绘图速度，可以使用底层(C/C++层)的OpenGL ES C版本。
- Android提供了一个基类来协助有上述需求的App开发者。
- 这个基类就是GLSurfaceView，它是基于SurfaceView框架，延伸而来的。



SurfaceHolder.Call

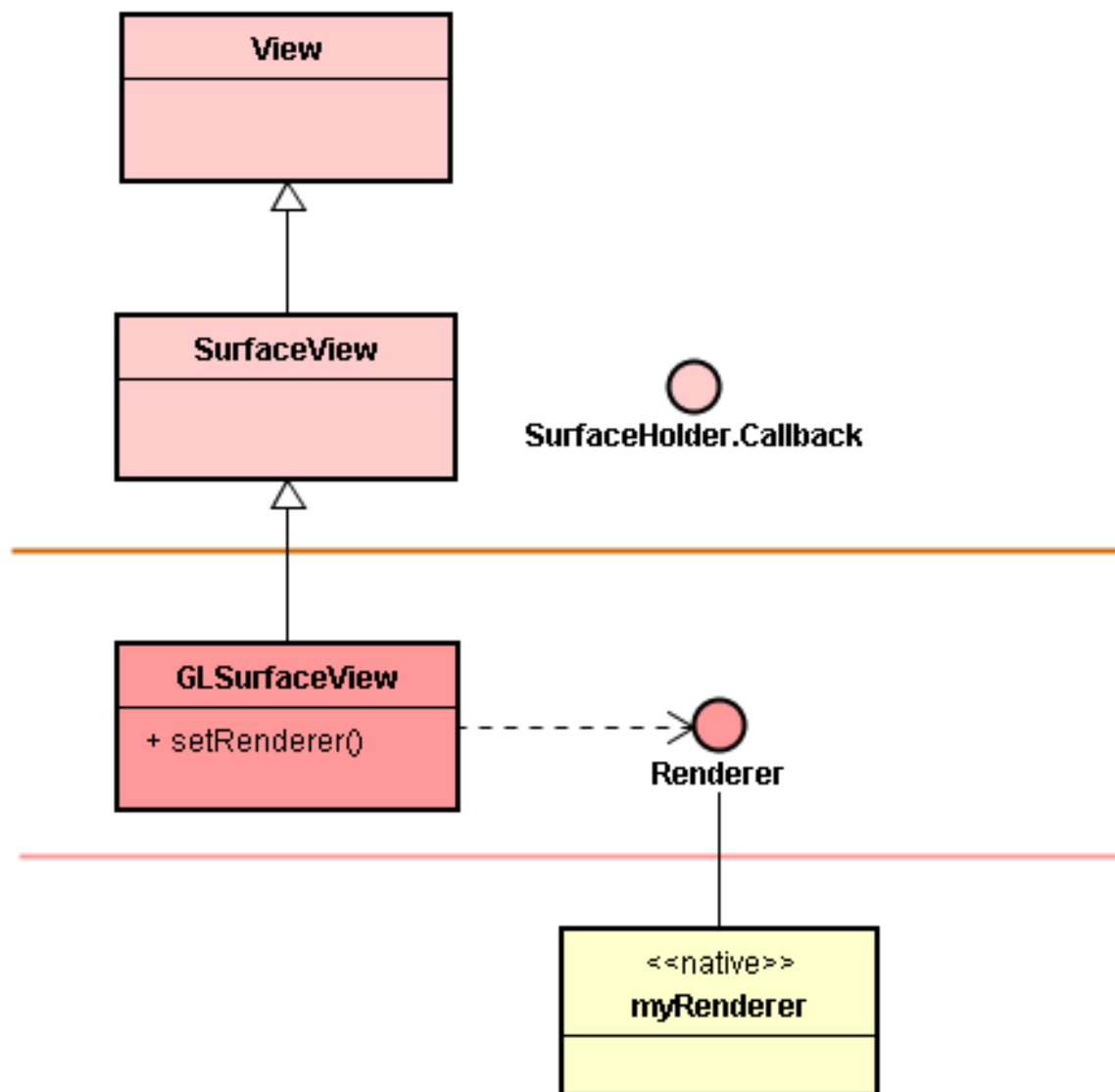
void onDrawFrame(...)
void onSurfaceChanged(...)
void onSurfaceCreated(...)

Renderer

特殊性框架

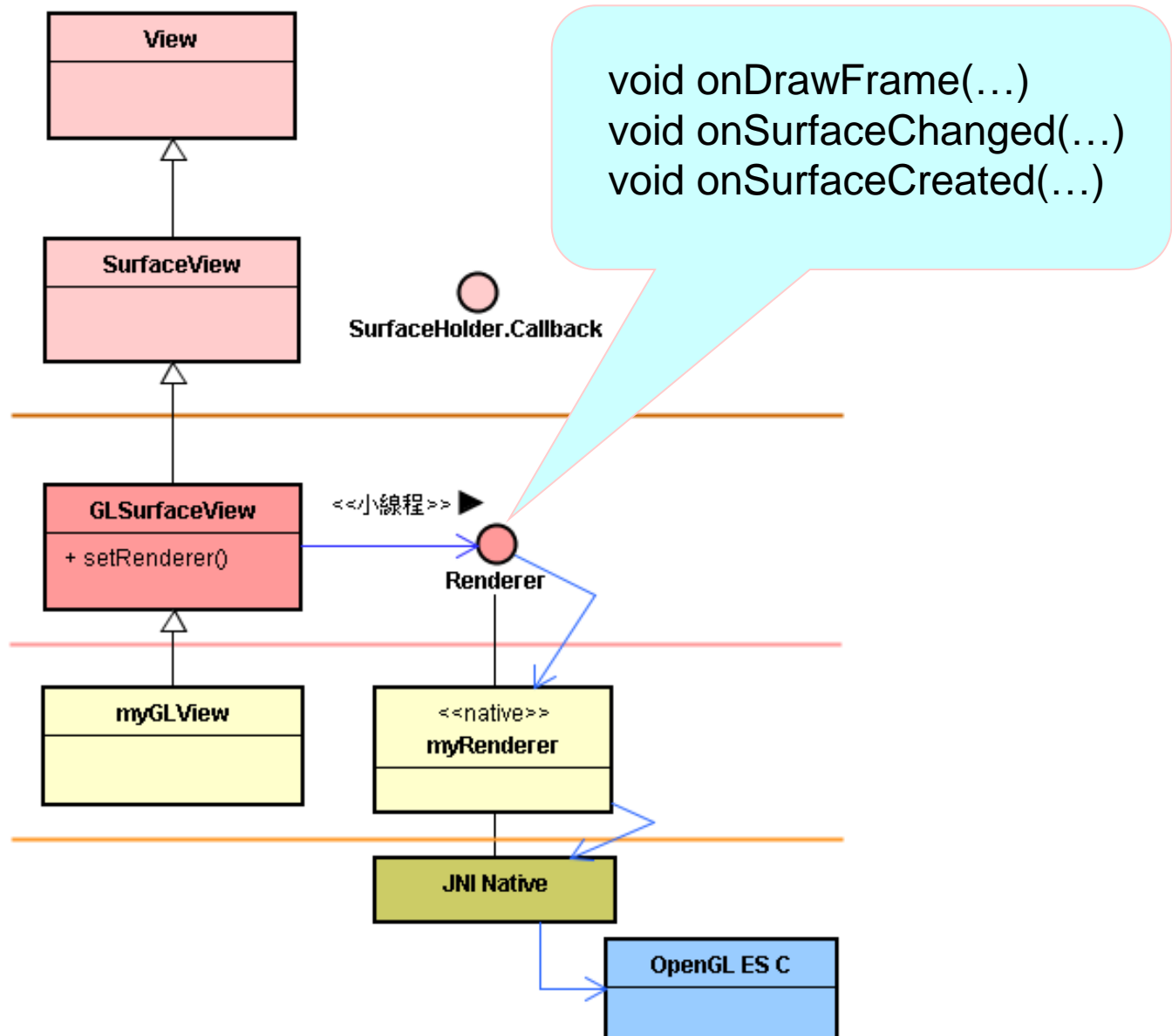
- 请注意，SurfaceView透过SurfaceHolder.Callback接口调用App的函数时，是使用UI线程的。
- 如果App想绘制3D动态图时，<T>自行诞生小线程去绘图。

- 然而，有了特殊性小框架，则由GLSurfaceView针对OpenGL特殊需求而来创建小线程。
- 所以，GLSurfaceView透过Renderer接口调用App的函数时，是使用小线程的。
- 基于上述的Renderer接口，就能撰写<T>来实作此接口，并与OpenGL引擎对接。



也能与C/C++层的
OpenGL引擎对接

- 此图里的myRenderer是Java层的类，它必须透过 JNI接口来与C/C++层的OpenGL ES C绘图引擎来沟通。如下图：



- 依循上图的规划，在装配阶段，可由myActivity来创建myGLView对象。
- 此myGLView对象再创建myRenderer对象。
- 然后，调用GLSurfaceView的setRenderer()函数来让GLSurfaceView指向myRenderer的Renderer接口。

- 建构好了，执行时，GLSurfaceView就透过Renderer接口来调用myRenderer的函数，将画布(即Surface)传递到下层，再经由JNI接口来让层的OpenGL ES C引擎进行动态的3D绘图了。



~ Continued ~