

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>

C01_a

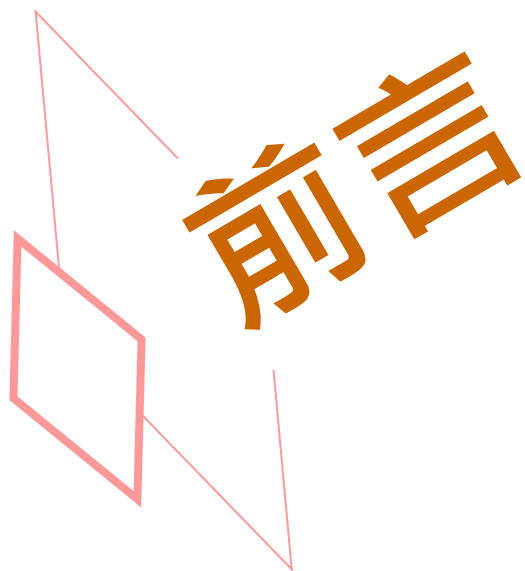
JNI架构原理： Java与C的对接(a)

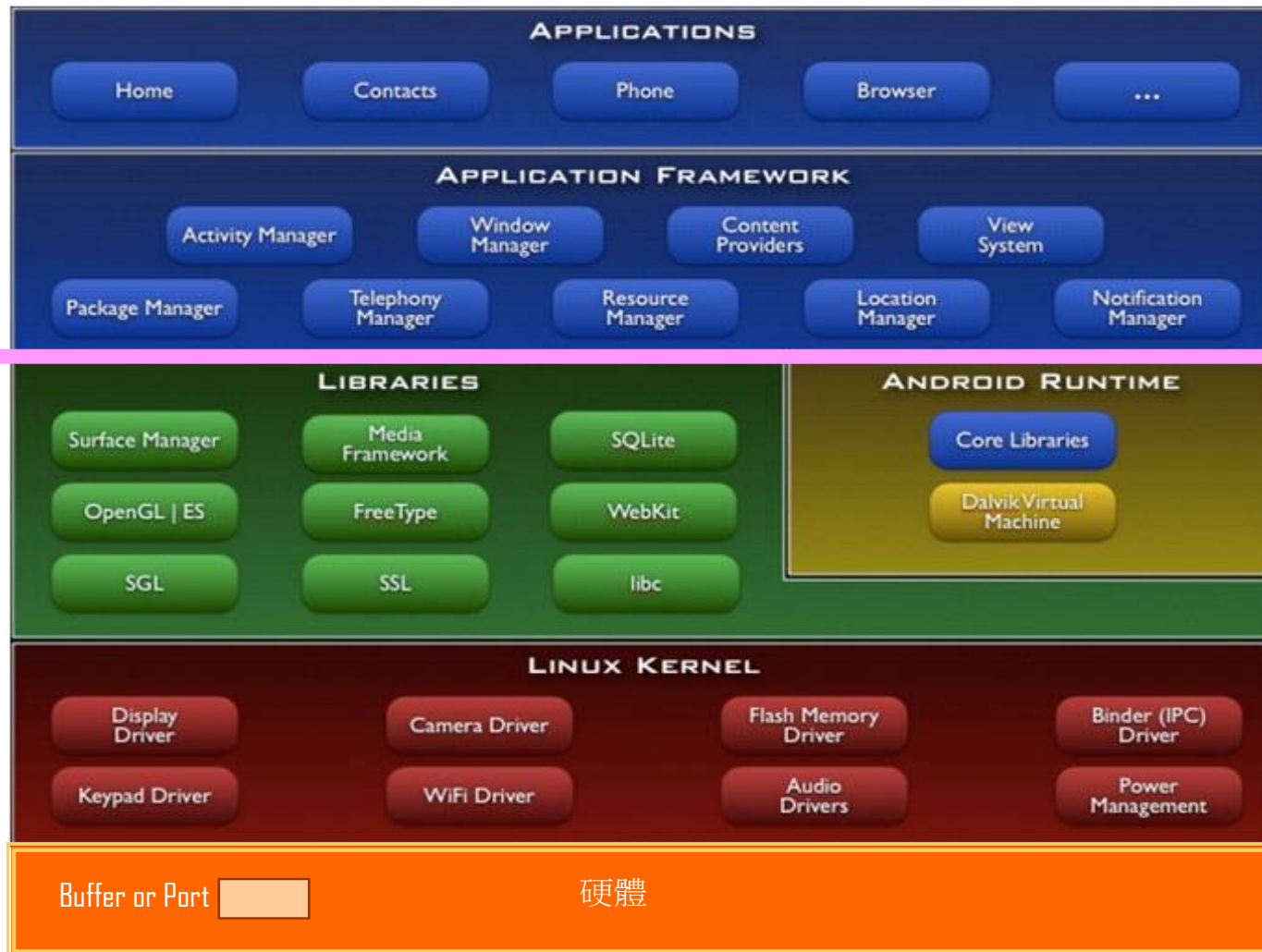
By 高煥堂

内容

1. 为什么，Android应用需要Java和C对接呢？
2. EIT造形的Java实现
3. EIT造形的C语言实现
4. EIT造形的C和Java组合实现

1、为什么，Android应用需要Java和C对接呢？





JNI

回顾与展望

Java的數據(對象)觀點

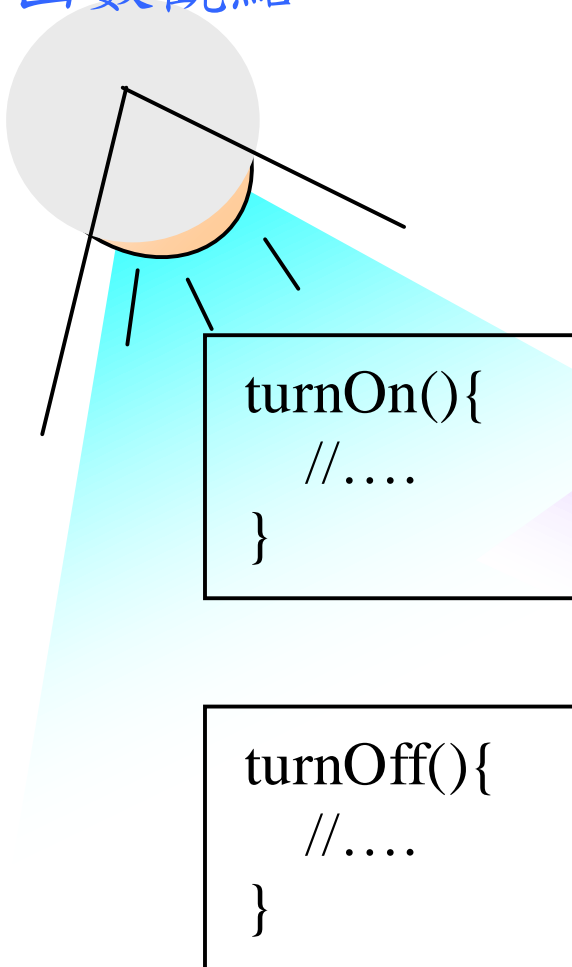
- 1970年代，以函数为中心的观点。

Ex. C, Pascal, ADA

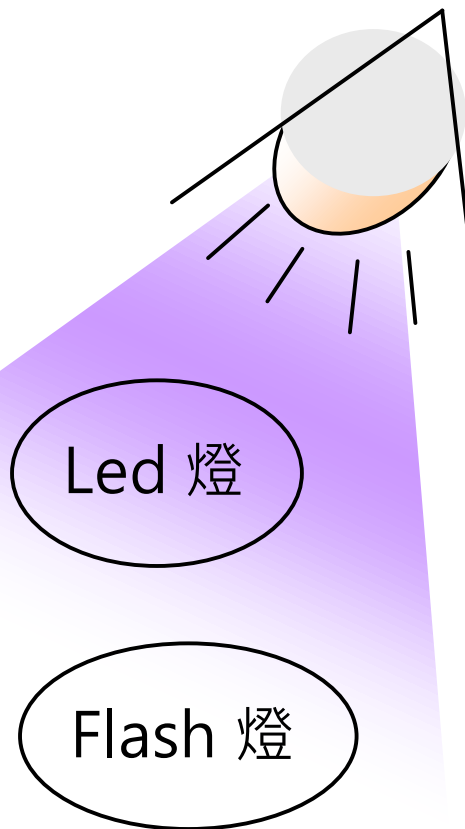
- 1980年代，以数据为中心的观点。

Ex. C++, Smalltalk, Objective-C

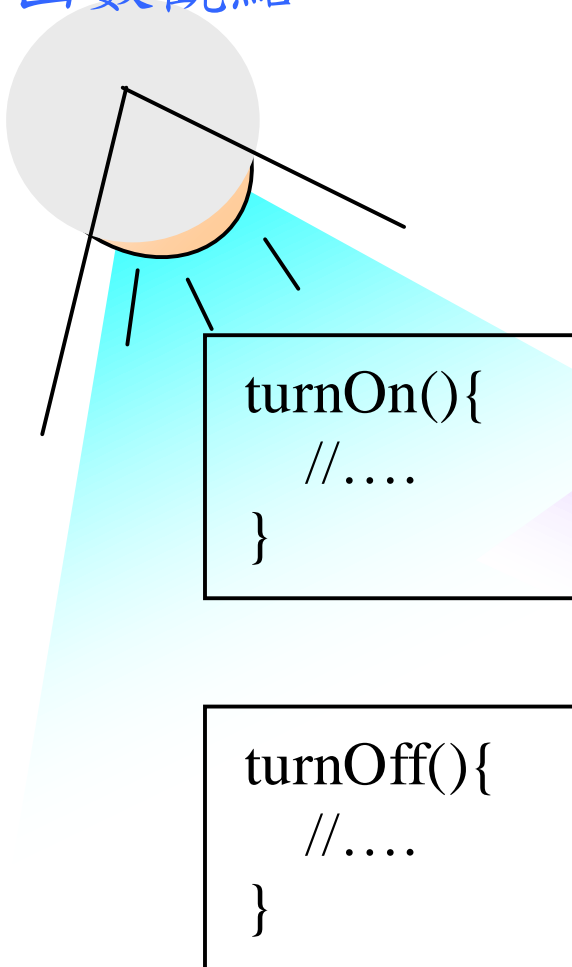
函數觀點



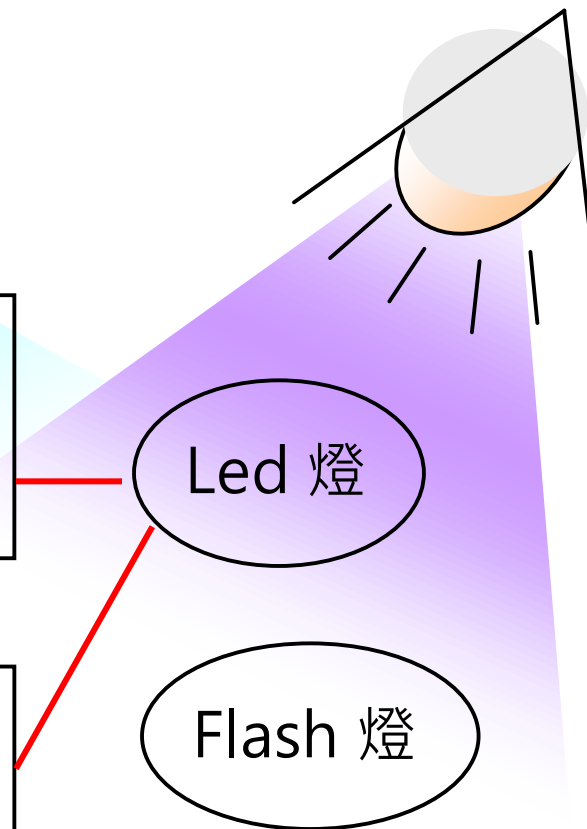
數據觀點



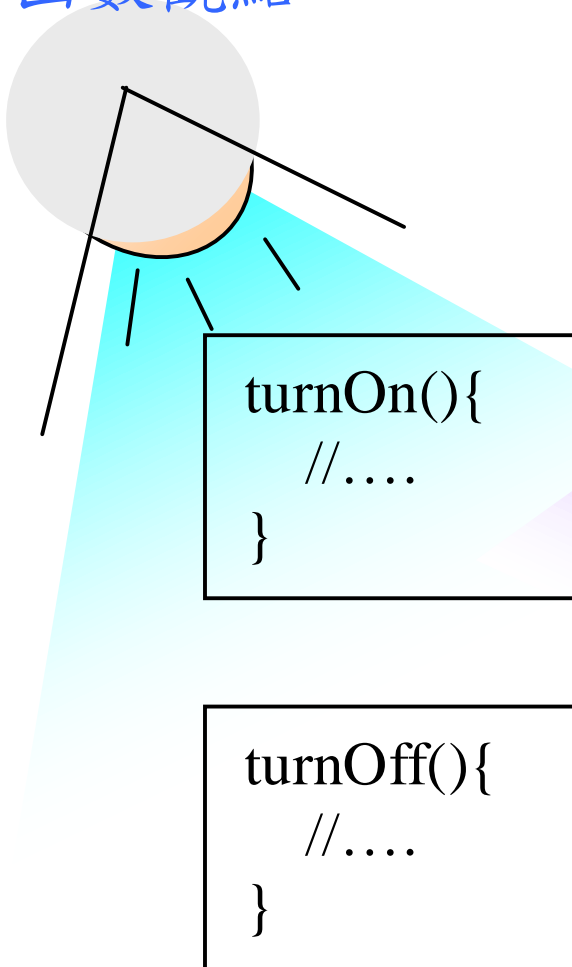
函數觀點



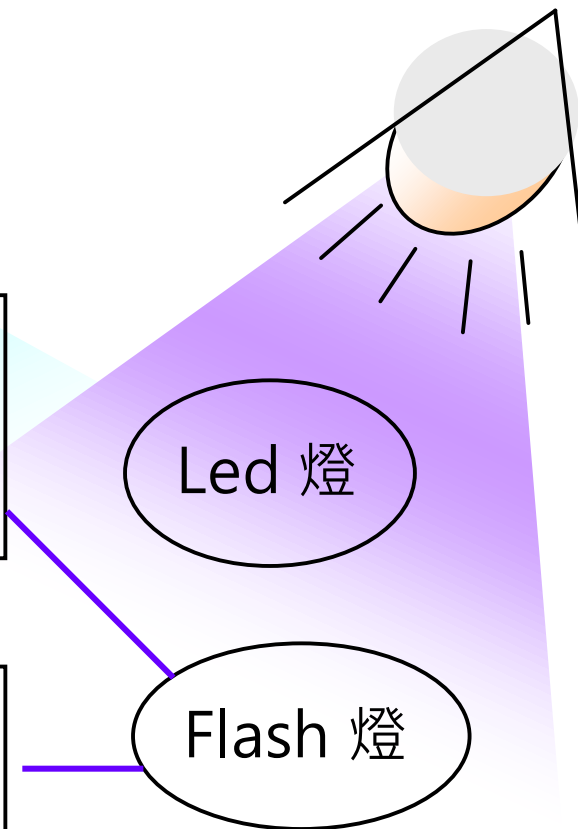
數據觀點



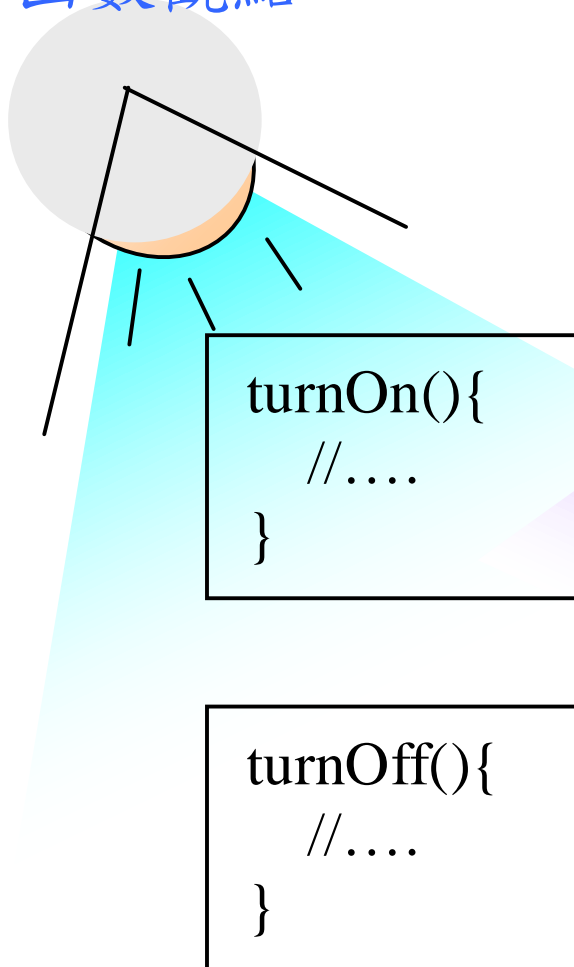
函數觀點



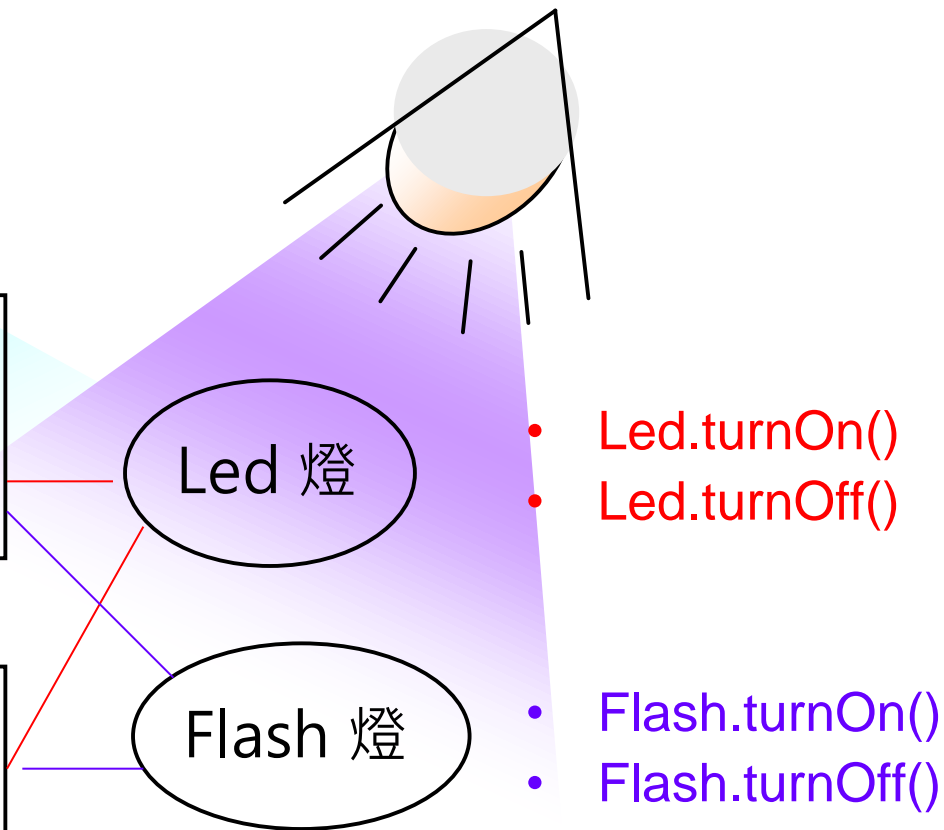
數據觀點



函數觀點



數據觀點



```
class Light {  
    int state;  
    void turnOn(){  
        state = 1;  
    }  
    void turnOff(){  
        state = 0;  
    }  
};
```

數據結構&函數

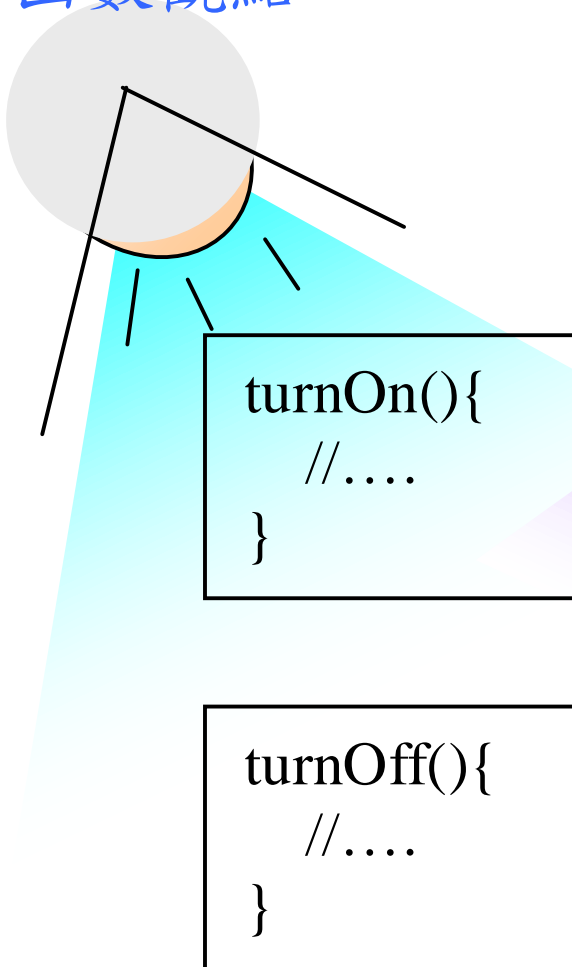
誕生對象&調用函數

```
Light Led = new Light();  
Led.turnOn();  
Led turnOff();
```

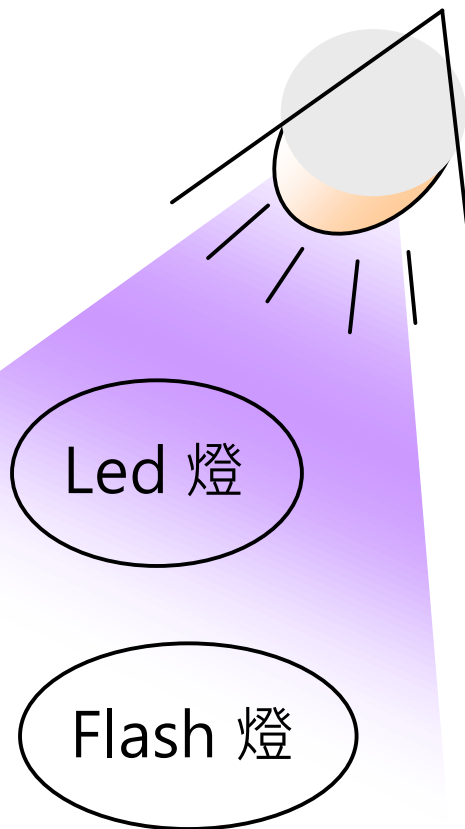
```
Light Flash = new Light();  
Flash.turnOn();  
Flash turnOff();
```

C的函數(程序)觀點

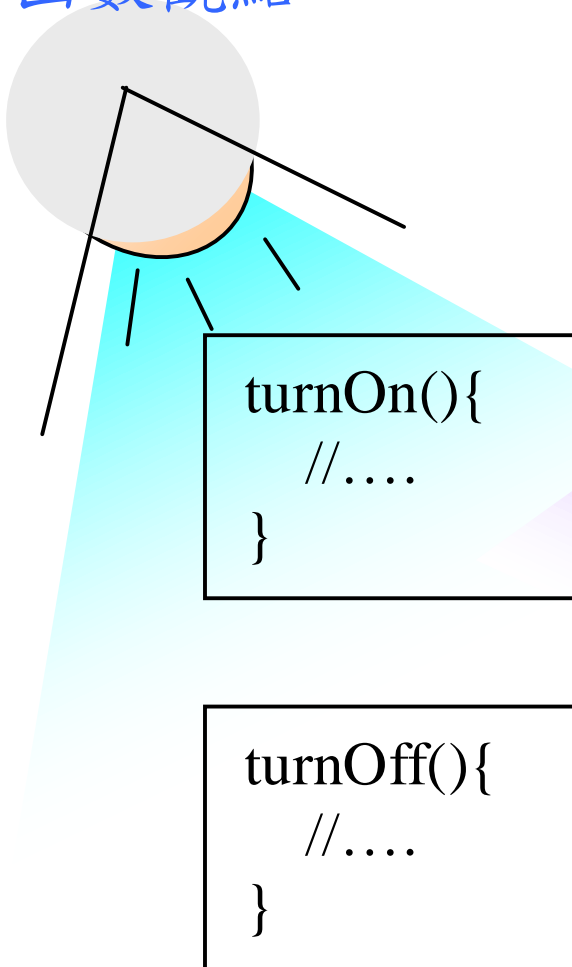
函數觀點



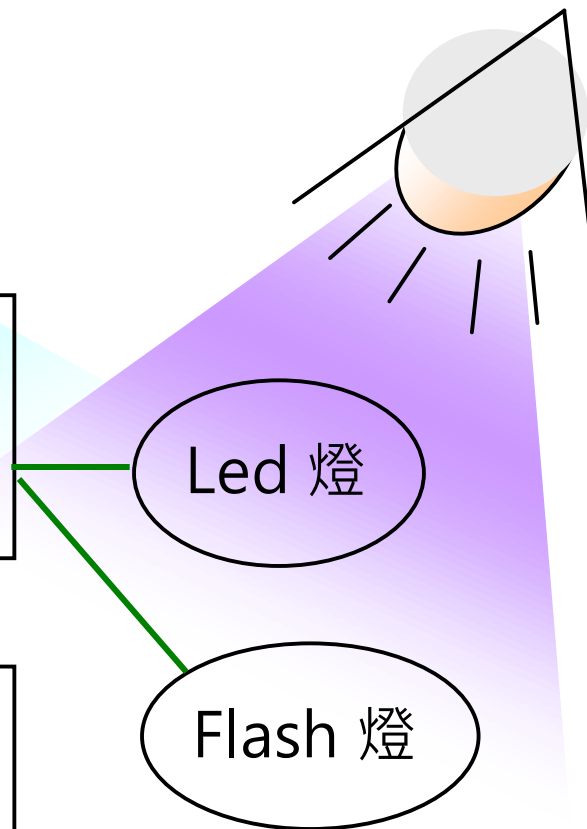
數據觀點



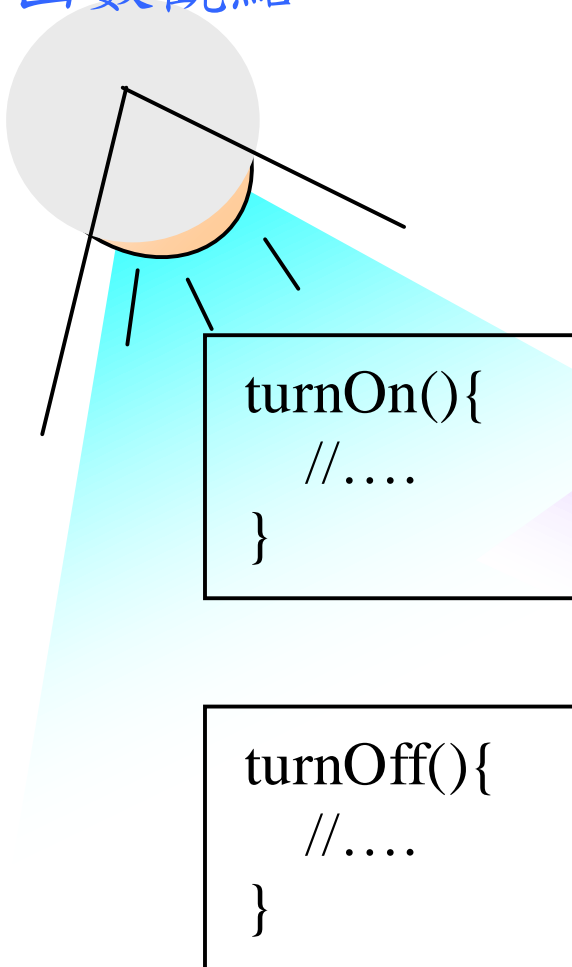
函數觀點



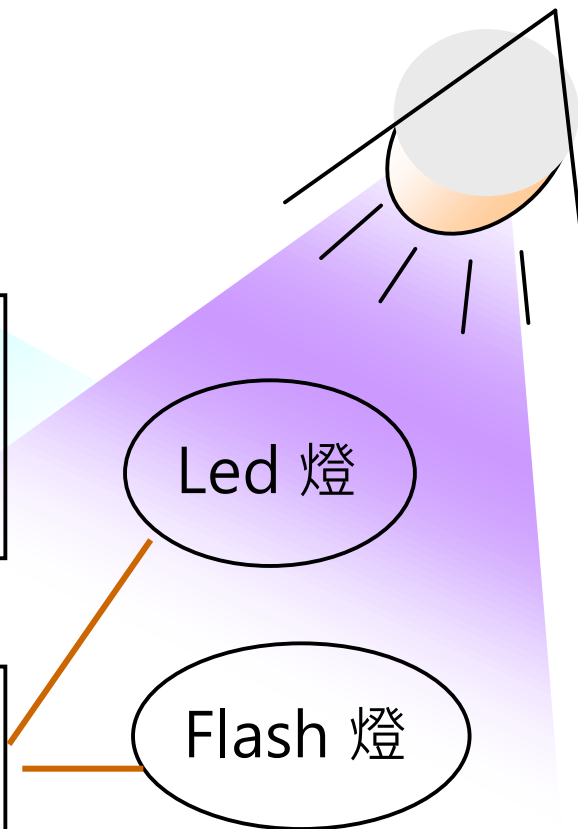
數據觀點



函數觀點



數據觀點



函數觀點

數據觀點

- turnOn(Led)
- turnOn(Flash)

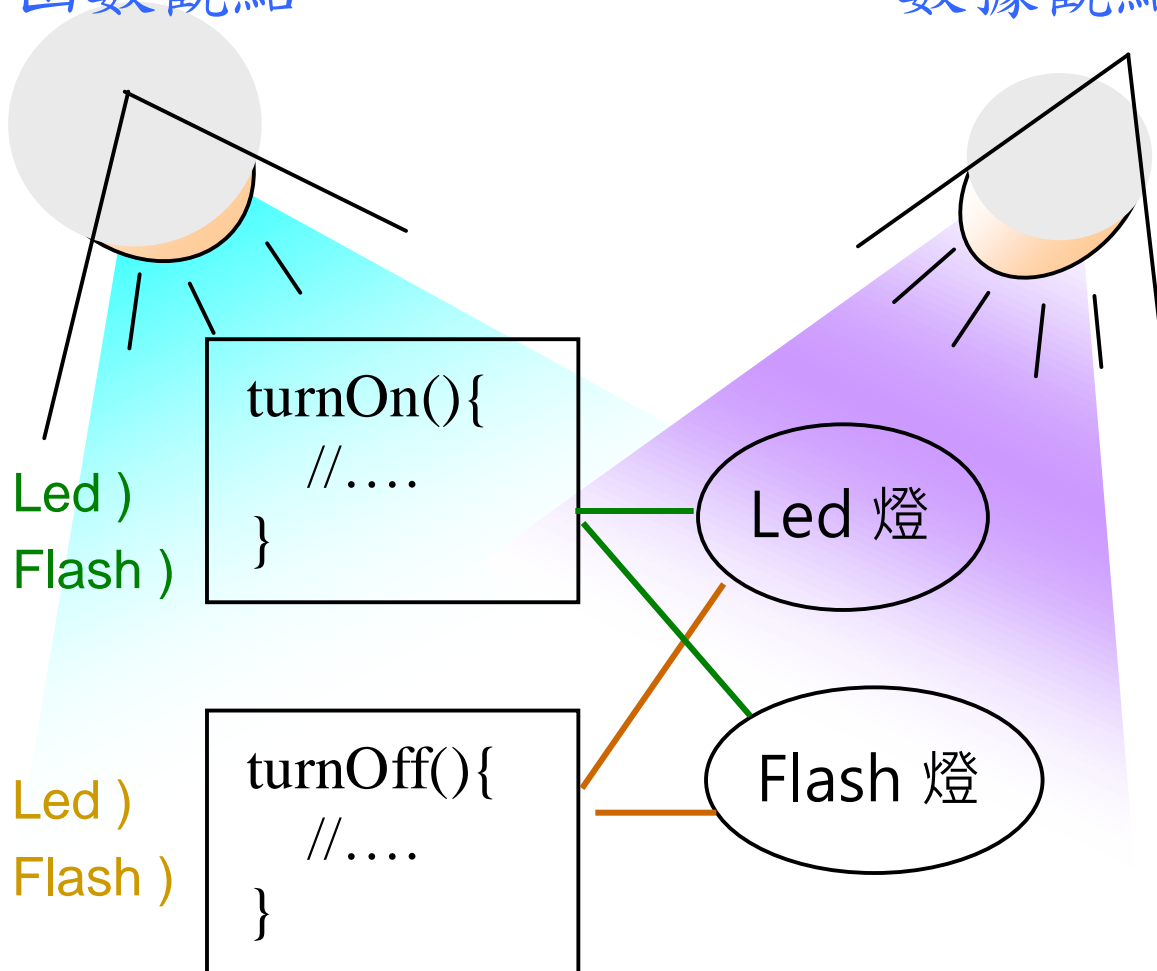
```
turnOn(){  
    //....  
}
```

- turnOff(Led)
- turnOff(Flash)

```
turnOff(){  
    //....  
}
```

Led 燈

Flash 燈



函數

```
// a.so 檔案(File)
void turnOn( Led *px )
{ px->state = 1; }
void turnOn(Flash *px )
{ px->state = 1; }
void turnOff(Led* px)
{ px->state = 0; }
void turnOff(Flash* px)
{ px->state = 0; }
```

數據結構

```
struct Light {
    int state;
};
```

誕生對象&調用函數

```
Led = malloc(...);
Light = malloc(...);
turnOn(Led);
turnOn(Flash);

turnOn(Led);
turnOn(Flash);
```

數據結構

// C代碼

函數

// a.so 檔案(File)

```
void turnOn( Led *px )  
    { px->state = 1; }  
void turnOn(Flash *px )  
    { px->state = 1; }  
void turnOff(Led* px)  
    { px->state = 0; }  
void turnOff(Flash* px)  
    { px->state = 0; }
```

誕生對象&調用函數

// C代碼

數據結構

// Java代碼

函數

// a.so 檔案(*File*)

```
void turnOn( Led *px )  
    { px->state = 1; }  
void turnOn(Flash *px )  
    { px->state = 1; }  
void turnOff(Led* px)  
    { px->state = 0; }  
void turnOff(Flash* px)  
    { px->state = 0; }
```

誕生對象&調用函數

// Java代碼

數據結構

JNI

C函數

```
// a.so 檔案(File)
void turnOn( Led *px )
{ px->state = 1; }
void turnOn(Flash *px )
{ px->state = 1; }
void turnOff(Led* px)
{ px->state = 0; }
void turnOff(Flash* px)
{ px->state = 0; }
```

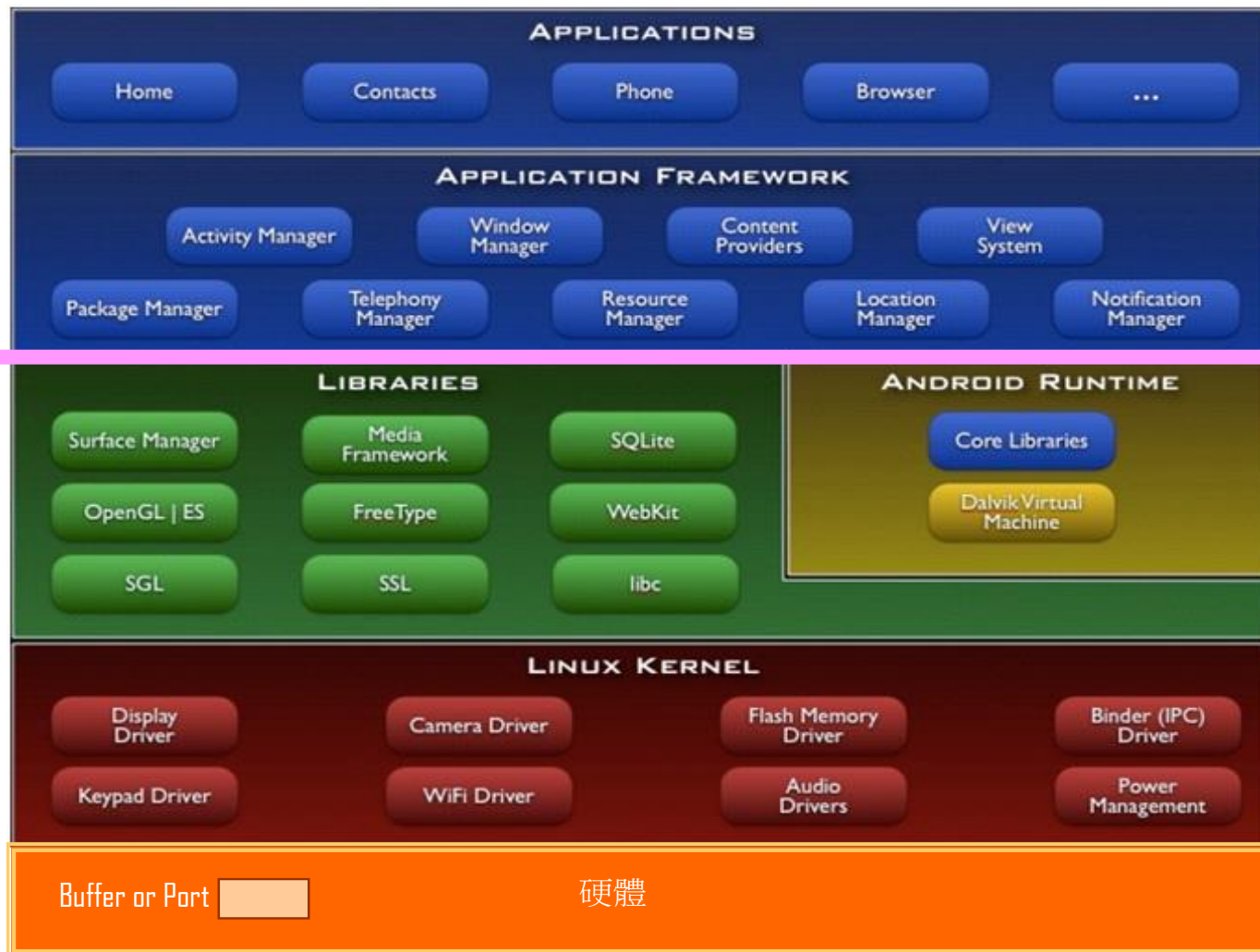
// Java代碼

誕生對象&調用函數

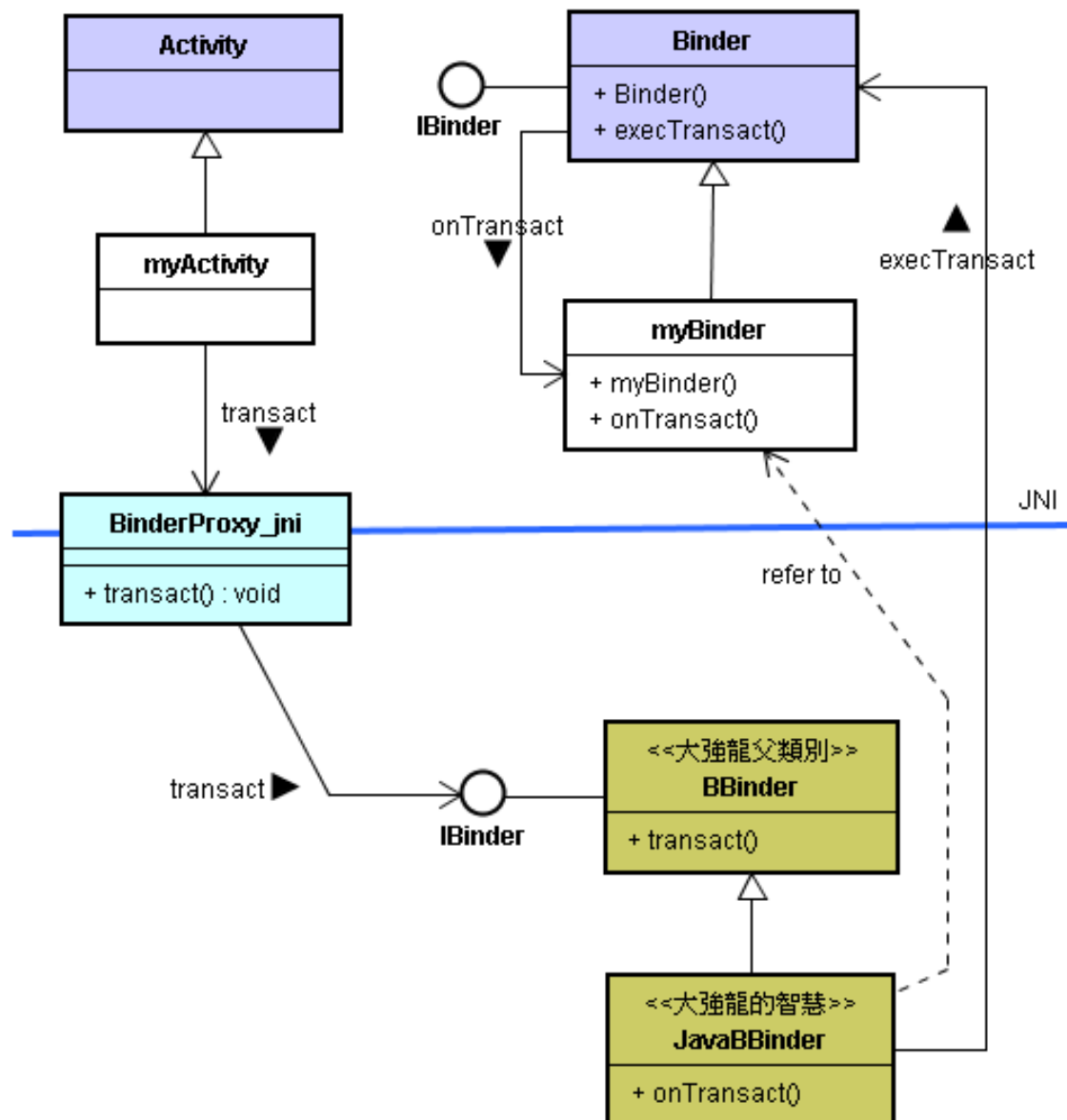
// Java代碼

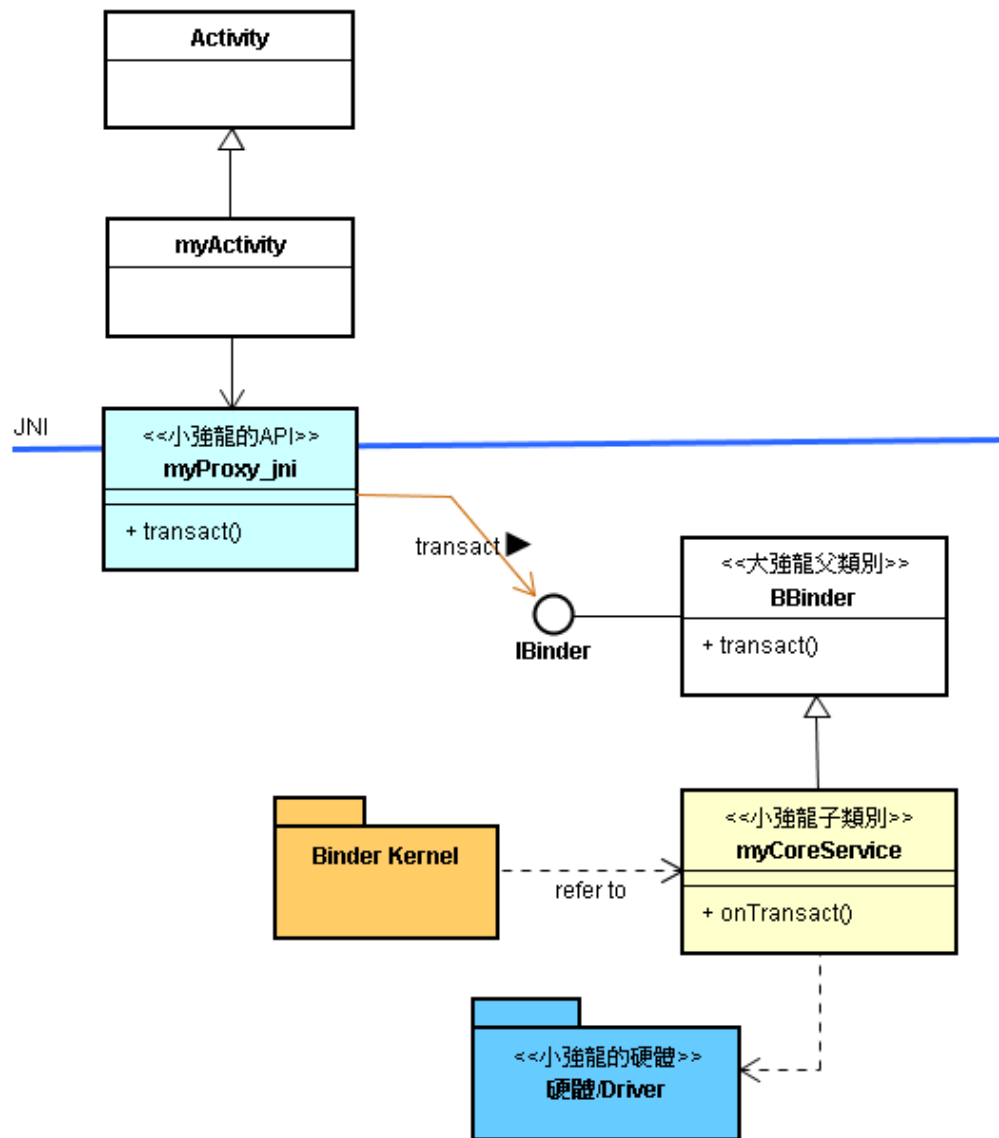
- 于是，Java与C就对接起来了。

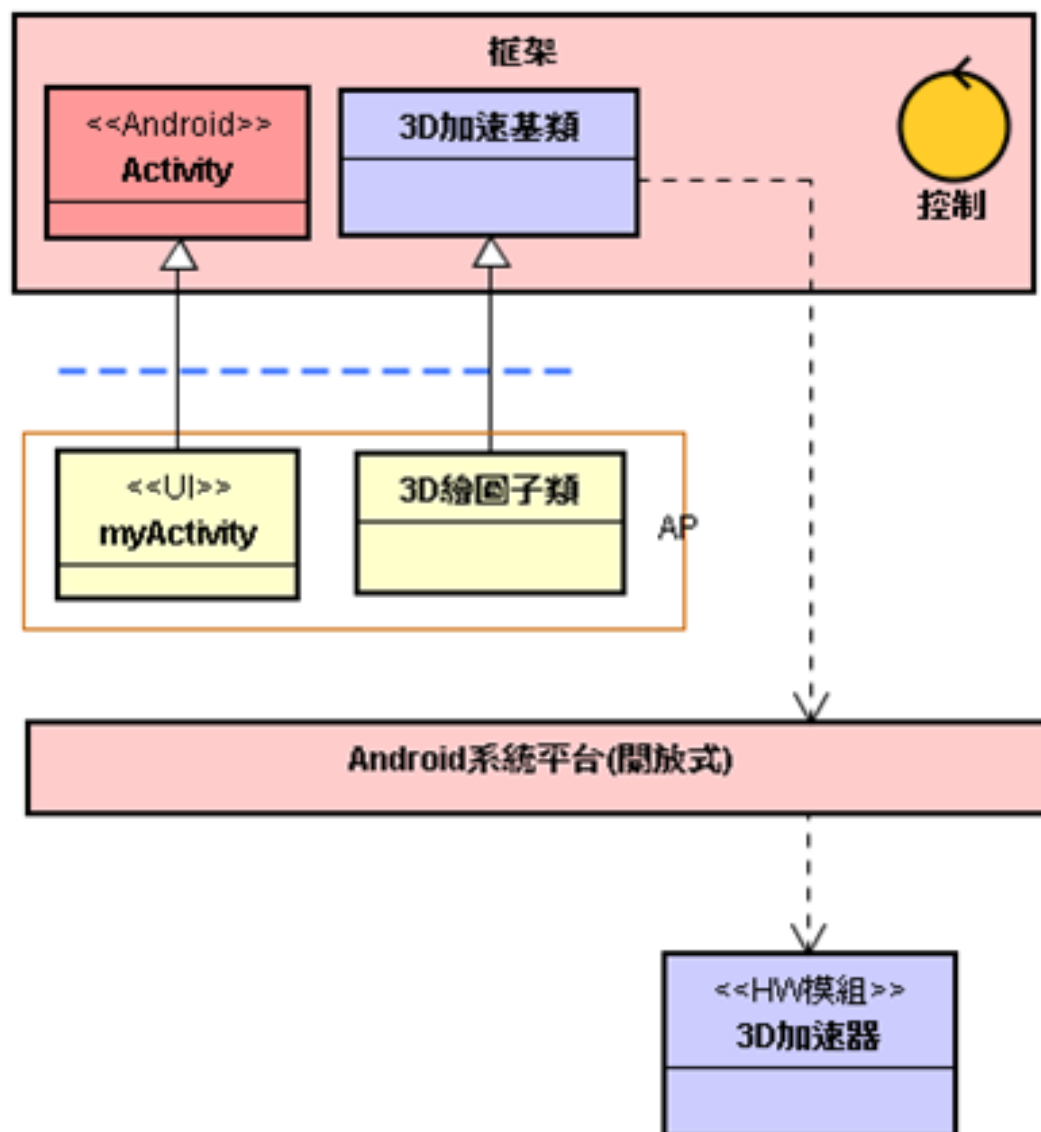
- 学习Android时，为什么也要学C/C++呢？
- Android的应用程序(App)可以完全以Java来撰写吗？
- 以C/C++撰写代码，是否比Java更容易于保护软件开发者的利益？

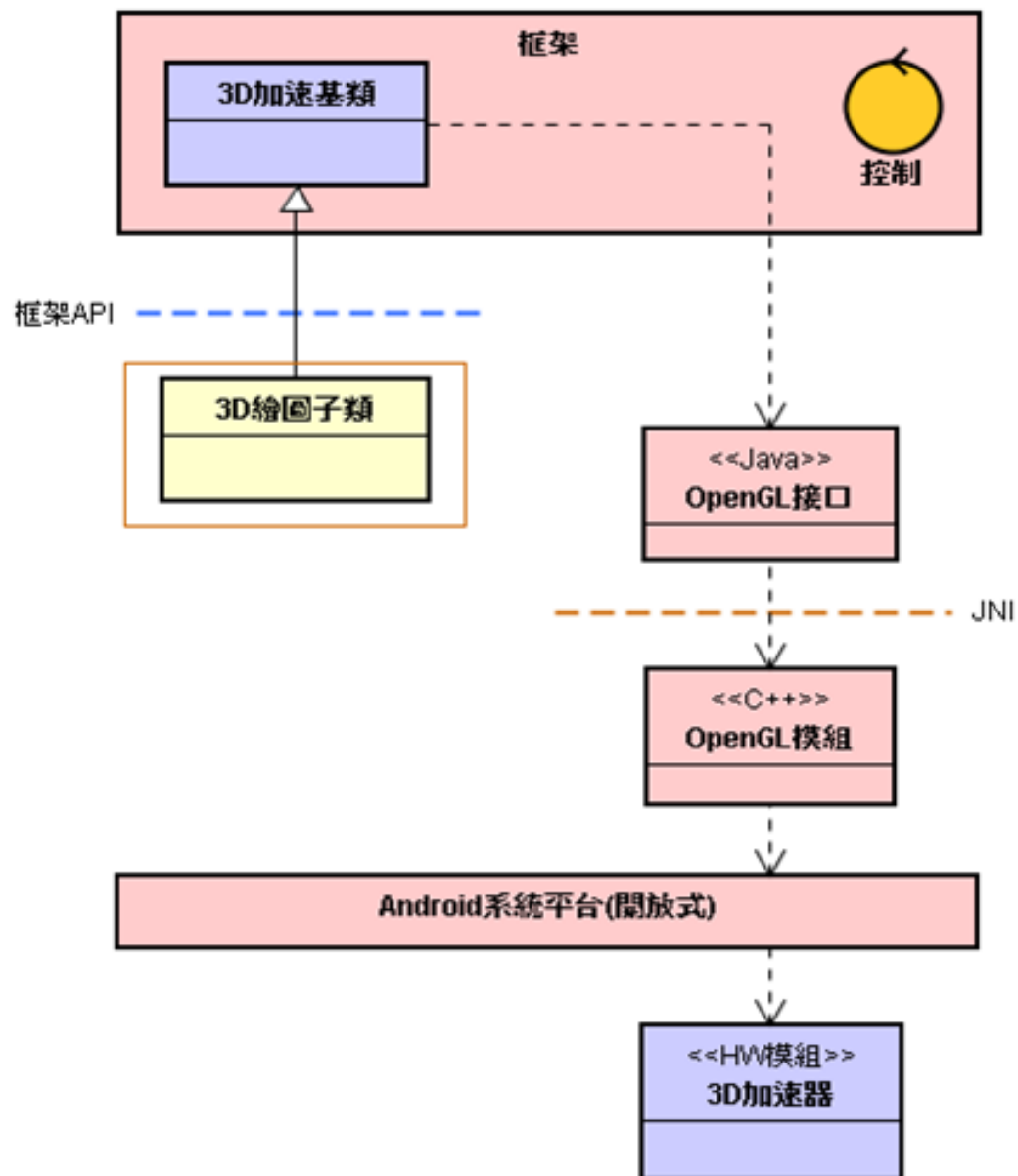


JNI











~ Continued ~