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

D05_c

Android Service的Proxy-Stub设计模式(c)

By 高煥堂

4、举例：以既有的SensorService为例

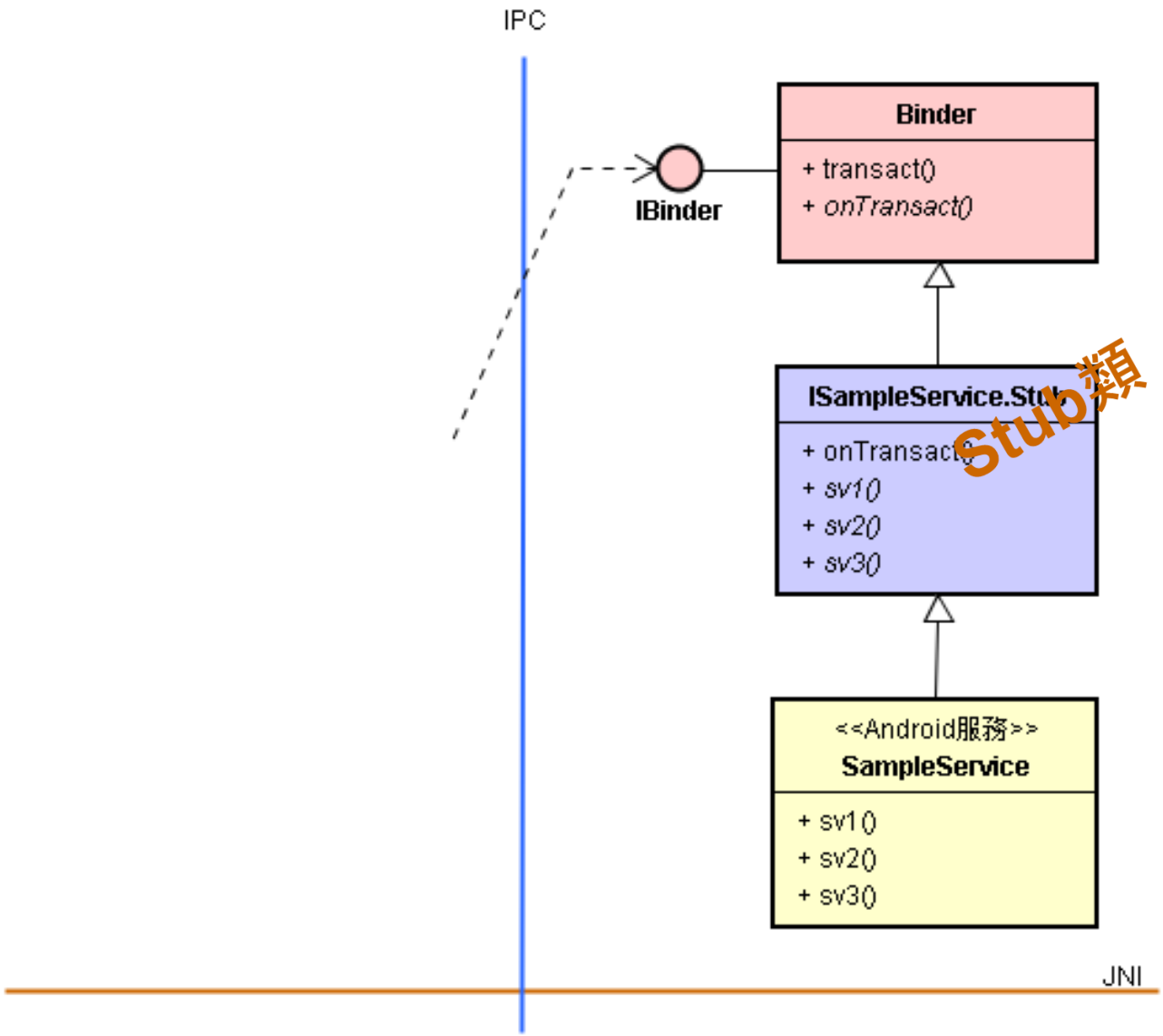
它的Stub和Proxy类

前言

- 从上一节里，可以看出使用AIDL来(生成Stub类)包装Android Service的要点：

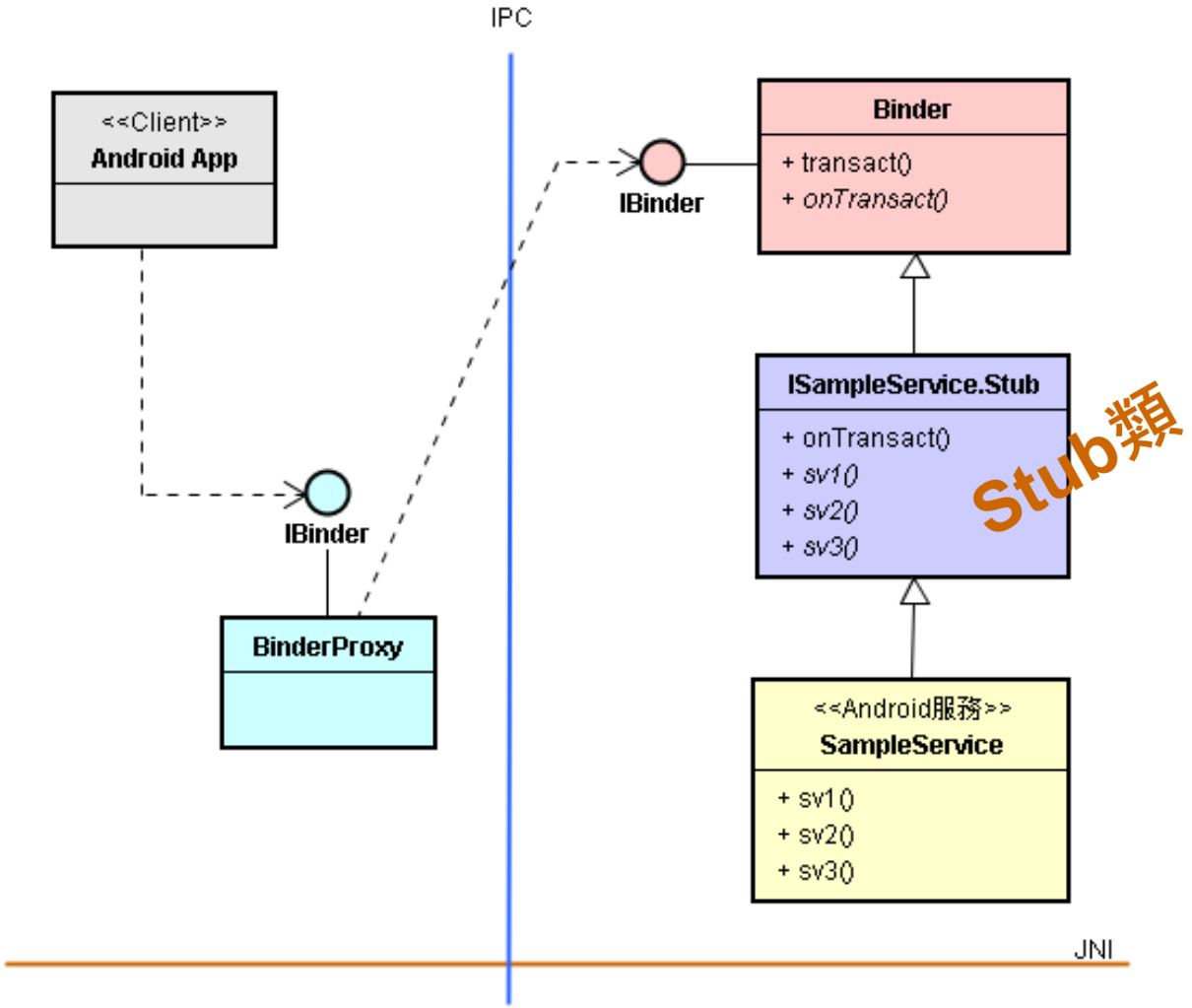
1. 地头蛇(服务开发者)定义一个ISampleService接口。

2. 使用强龙提供的AIDL工具去读取接口定义檔，产生一个Stub类和Proxy类；这个Stub类就是：ISampleService.Stub类。
3. 然后，撰写一个SampleService类来继承ISampleService.Stub基类。

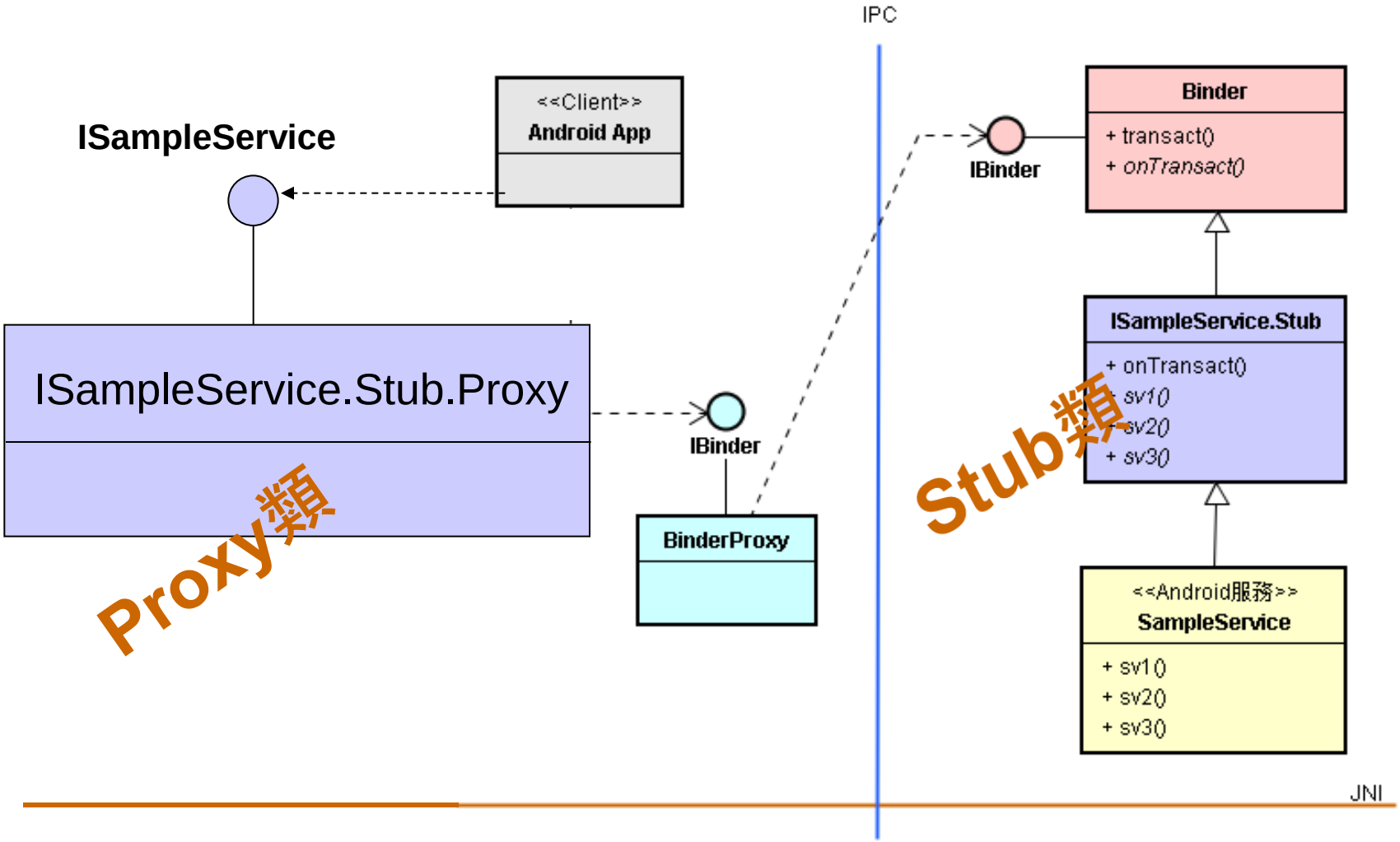

IPC
Binder
+ transact()
+ onTransact()
IBinder
ISampleService.Stub
+ onTransact()
+ sv1()
+ sv2()
+ sv3()
Stub類
<<Android服務>>
SampleService
+ sv1()
+ sv2()
+ sv3()
JNI

```
public class myActivity extends Activity {
@Override
    public void onCreate(Bundle savedInstanceState) {
          // ……
         IBinder m_ib  =  ServiceManager.getService("CS001");
         ISampleService m_obj  =
                    ISampleService.Stub.asInterface( m_ib );
          // ……
          m_obj.f1(255);
          m_obj.f2(17.35);
          // ……
}
```

IBinder m_ib = ServiceManager.getService("CS001");



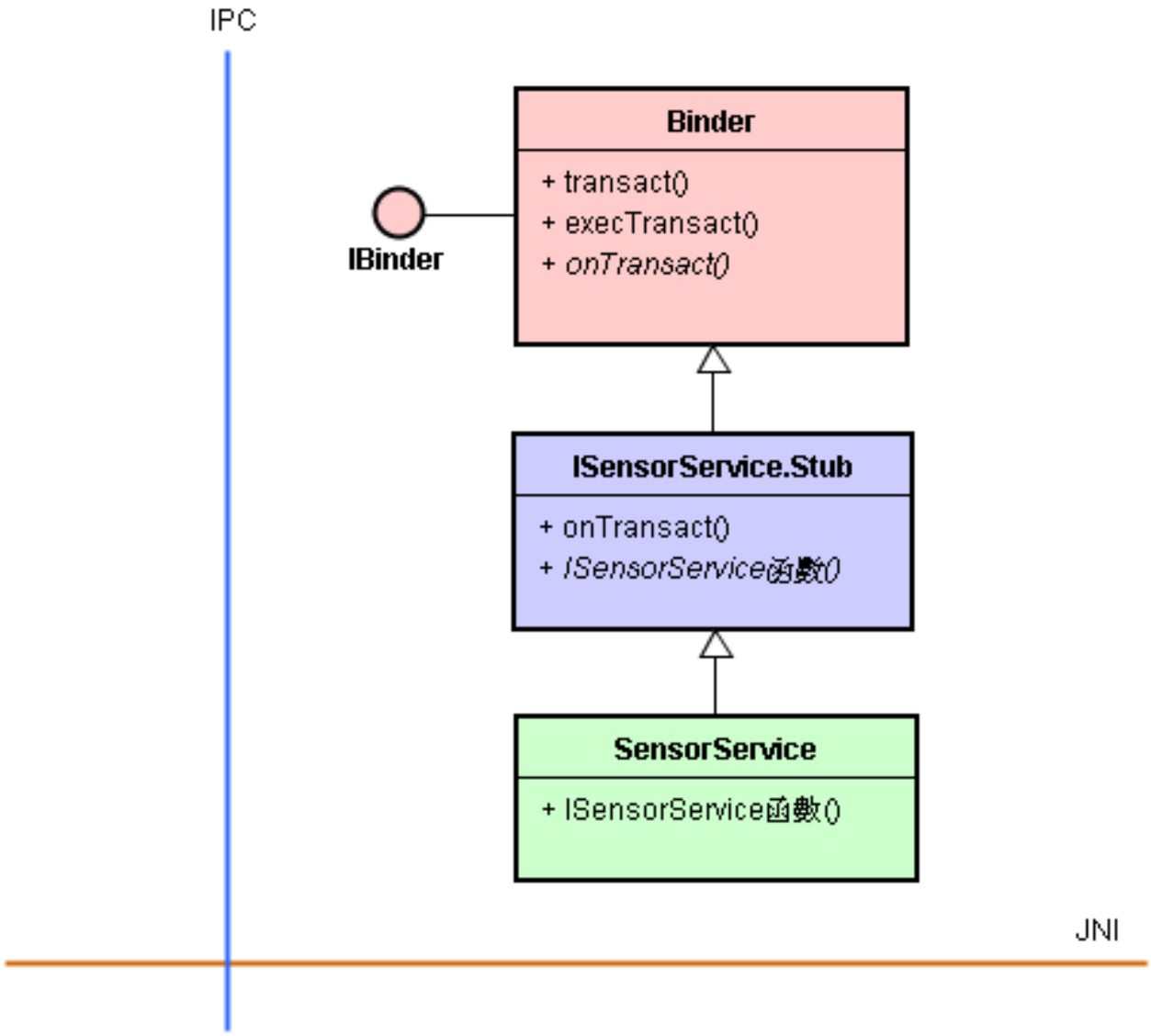
```
ISampleService m_obj  =
        ISampleService.Stub.asInterface( m_ib );
```



观摩ISensorService.Stub类

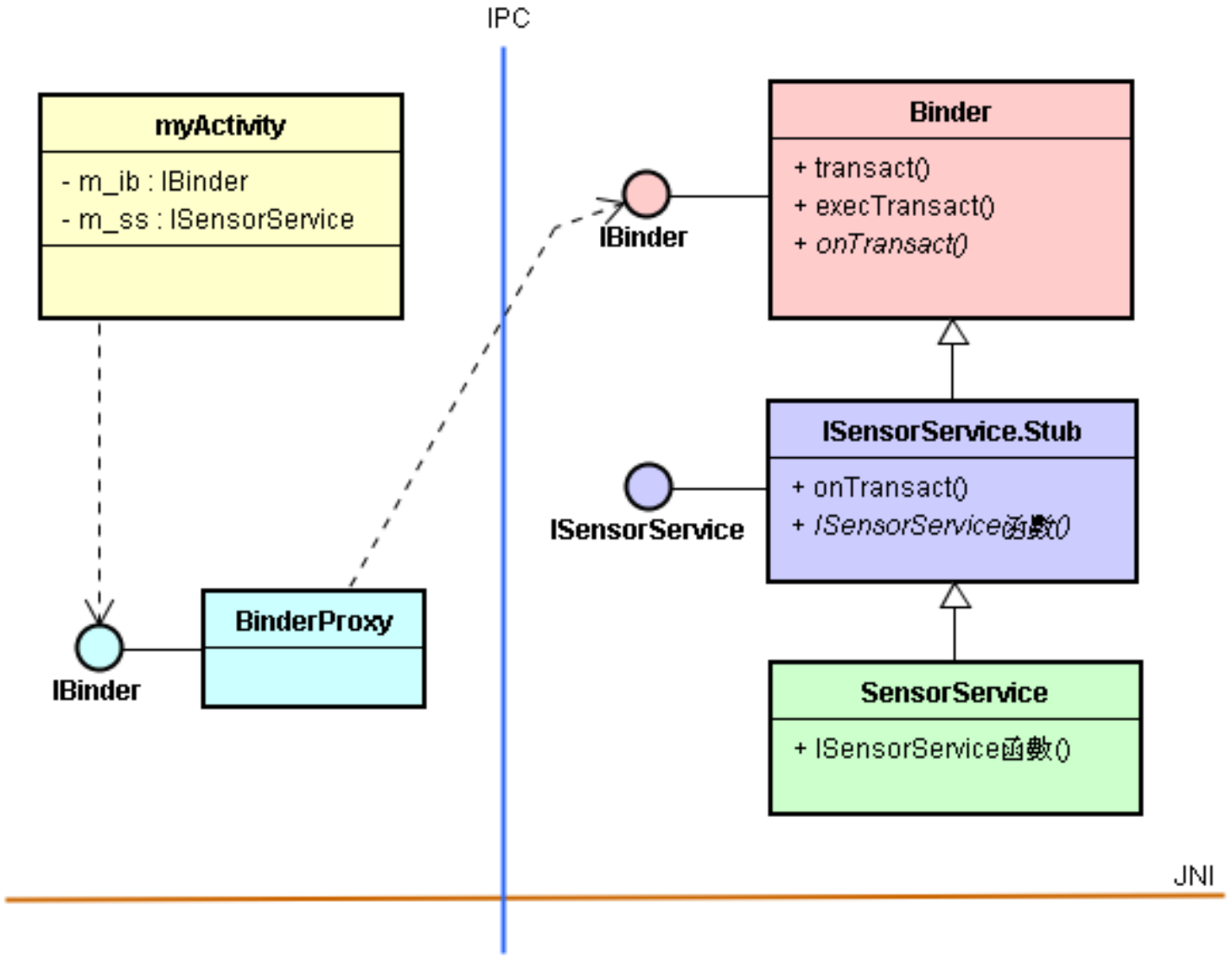
- 于是可得知，在Android (1.x和2.x版)的框架里，其SensorService类别就继承了ISensorService.Stub类，如下述代码：

```
// framework/base/services/java/SensorService.java
//-----------------------------------------------------------------
class SensorService extends ISensorService.Stub {
    //.................
    // (省略)
    //.................
    private static native int _sensors_control_init();
    private static native ParcelFileDescriptor _sensors_control_open();
    private static native boolean _sensors_control_activate(int sensor,
                        boolean activate);
    private static native int _sensors_control_set_delay(int ms);
    private static native int _sensors_control_wake();
}
```

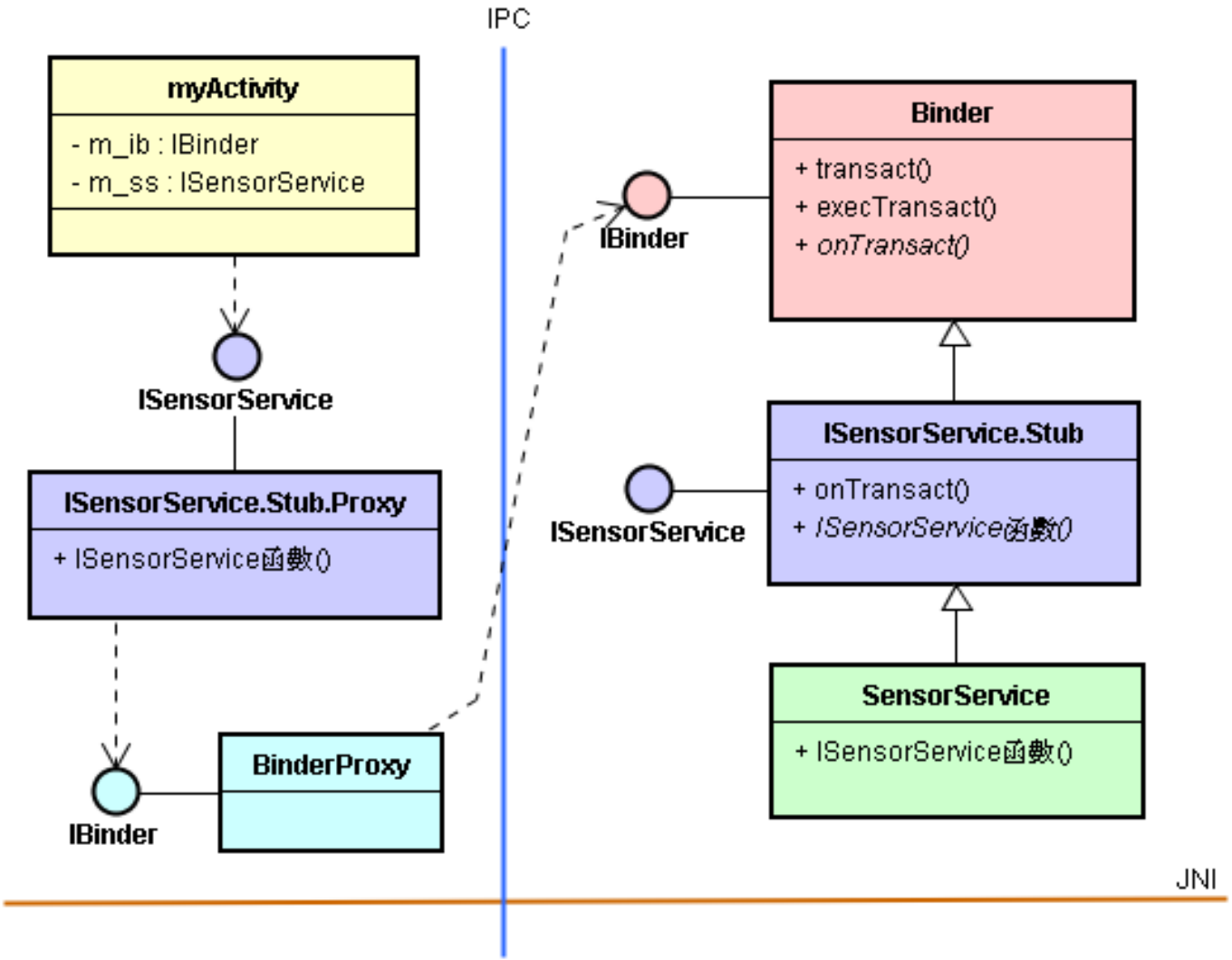

IPC
Binder
+ transact()
+ execTransact()
+ onTransact()
IBinder
ISensorService.Stub
+ onTransact()
+ ISensorService函數()
SensorService
+ ISensorService函數()
JNI

```
public class myActivity extends Activity {
    private static final String TAG="myActivity";

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        IBinder m_ib =  ServiceManager.getService(
                                    Context.SENSOR_SERVICE));
        ISensorService m_ss =
              ISensorService.Stub.asInterface( m_ib );
        // …….
    }
    // ……
}
```

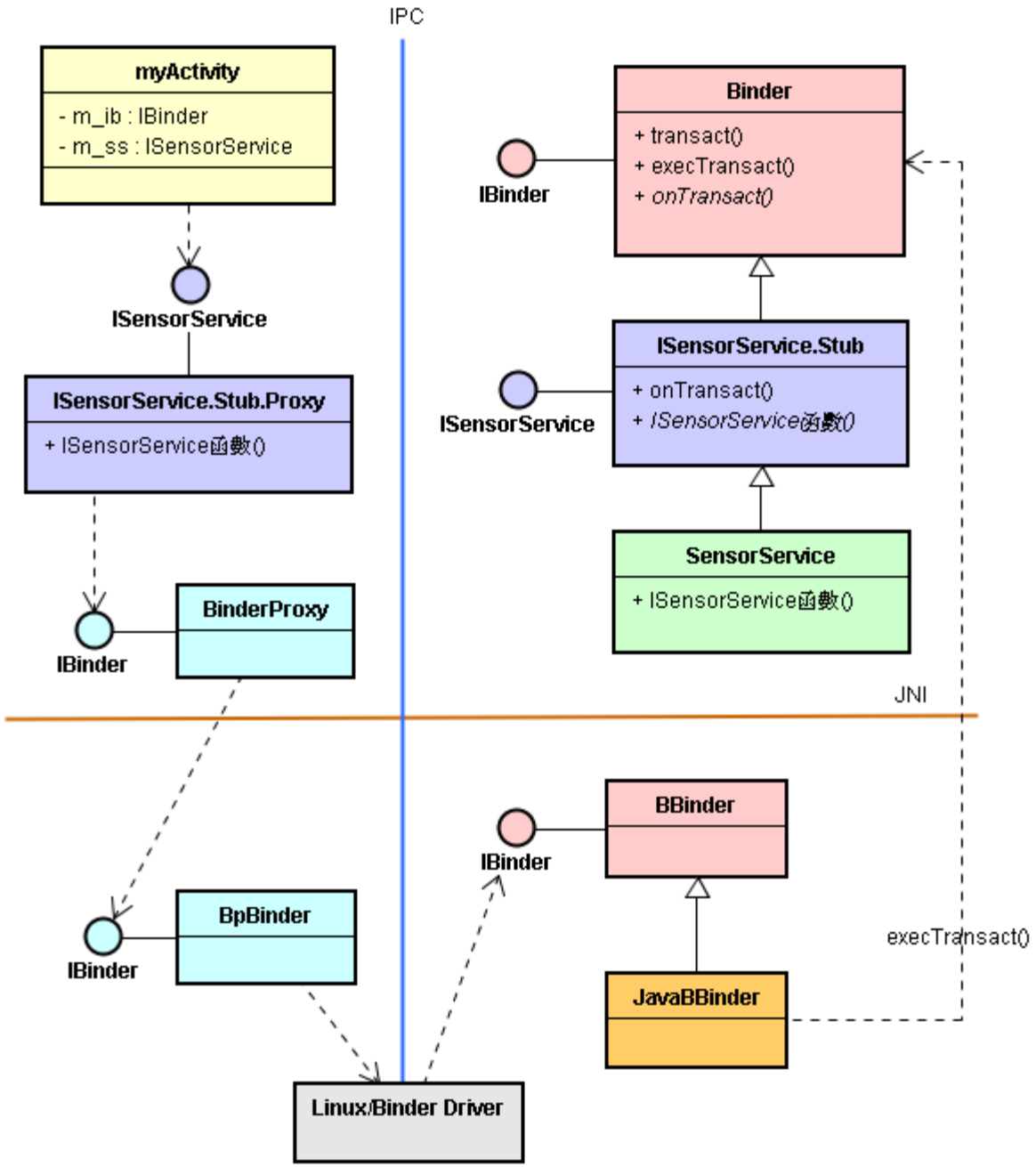
IBinder m_ib
= ServiceManager.getService(Context.SENSOR_SERVICE));



ISensorService m_ss =
ISensorService.Stub.asInterface(m_ib);



```
/**
 * Cast an IBinder object into an ISensorService interface,
 * generating a proxy if needed.
 */
public static
android.hardware.ISensorService asInterface(android.os.IBinder obj)
    {
        if ((obj==null))  return null;
        android.hardware.ISensorService in =
                android.hardware.ISensorService)obj.queryLocalInterface(
                                                DESCRIPTOR);
        if ((in!=null))  return in;
        return new android.hardware.ISensorService.Stub.Proxy(obj);
      }
```


IPC
myActivity
- m_ib : IBinder
- m_ss : ISensorService
ISensorService
ISensorService.Stub.Proxy
+ ISensorService函数()
IBinder
BinderProxy
Binder
+ transact()
+ execTransact()
+ onTransact()
IBinder
ISensorService.Stub
+ onTransact()
+ ISensorService函数()
ISensorService
SensorService
+ ISensorService函数()
JNI
IBinder
BpBinder
IBinder
BBinder
JavaBBinder
execTransact()
Linux/Binder Driver

Thanks...

高煥堂