

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>

D02_a

撰写你的第一个核心服务(a)

By 高煥堂

內容

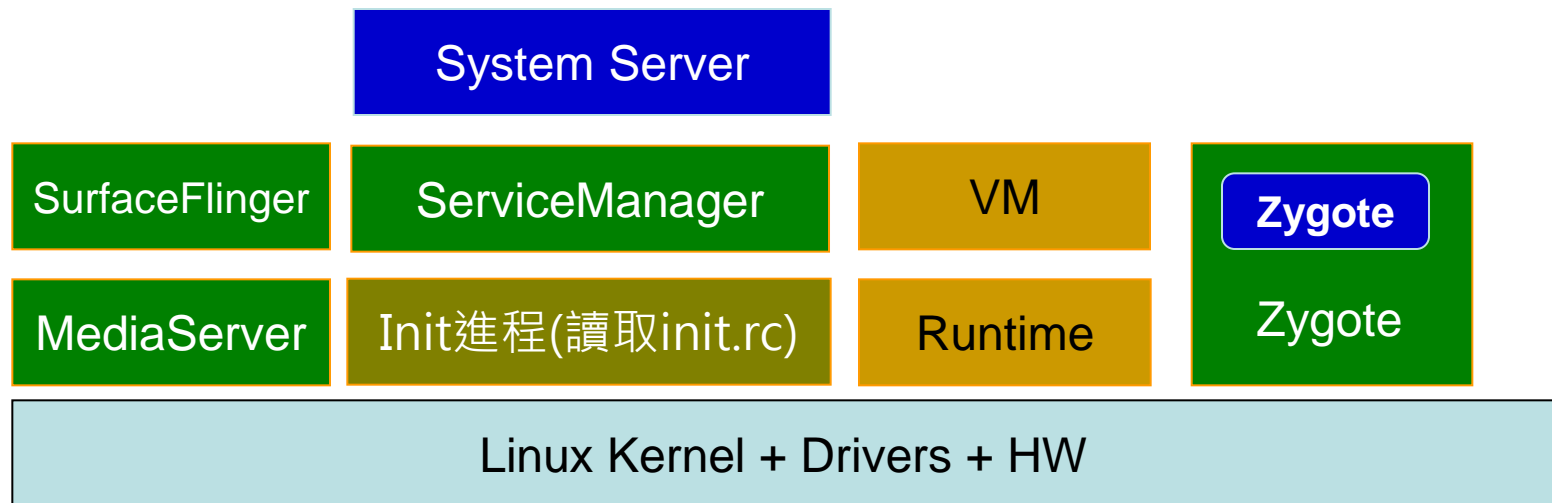
1. 为什么要写核心服务呢？
2. 亲自开发一个Native Service：SQRService
3. 撰写SQR.CPP 來使用SQRService
4. 撰写Java层Client來使用核心服务
5. 撰写一个Java层Activity
6. 优化设计

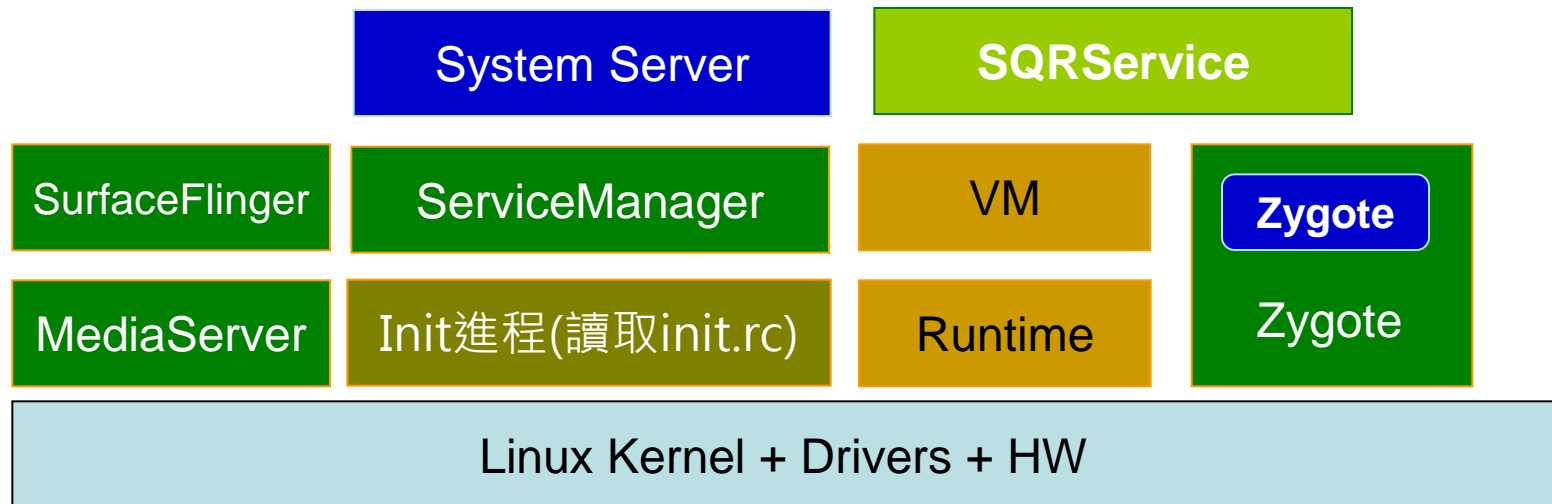
1、为什么要写核心服务呢？

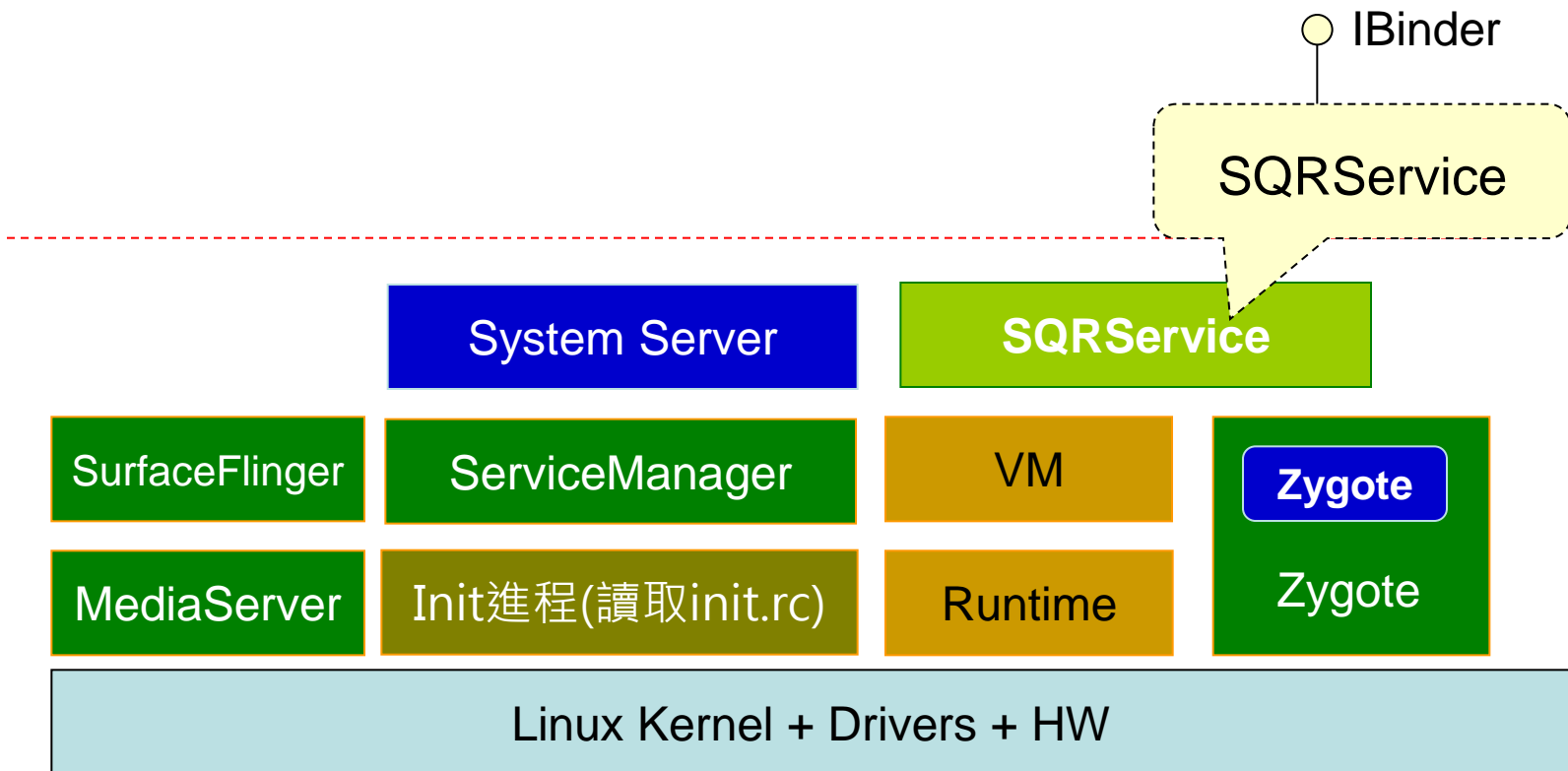
- 为什么需要学习撰写核心服务呢？其常见的理由是：
 1. 核心服务是 Android 框架裡最接近 Linux/Driver 的部分。为了充分发挥硬件设备的差异化特性，核心服务是让上层 Java 应用程序來使用 Driver/HW Device 特色的重要管道。

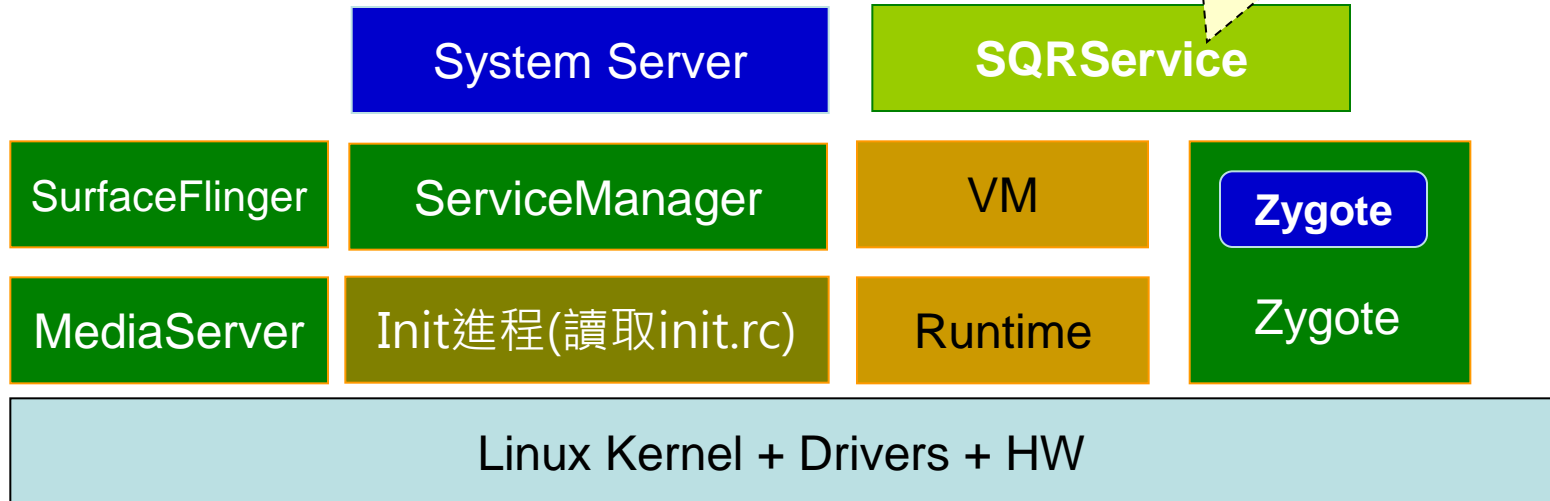
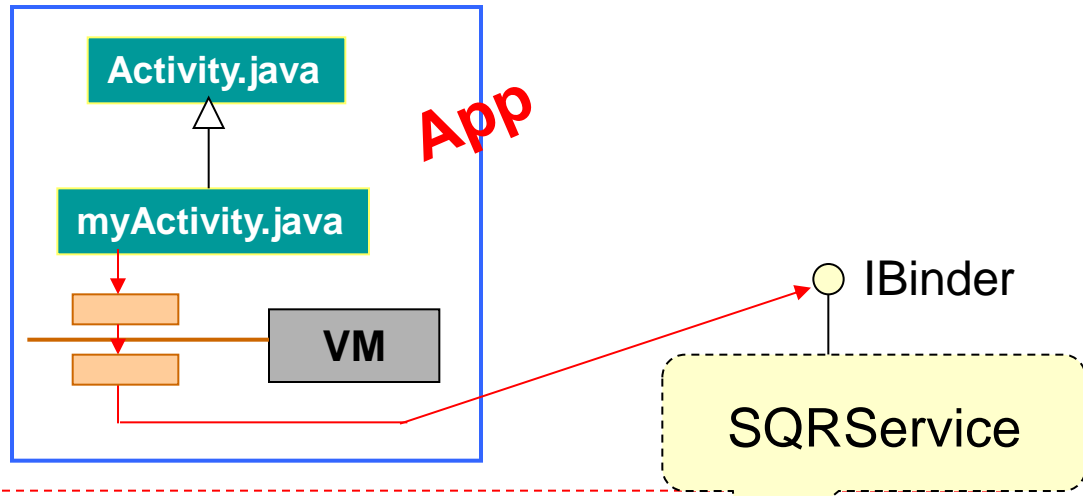
2. 在开机过程中，就可以启动核心服务(例如汉字输入法服务等)，让众多应用程序來共享之。
3. 由于共享，所以能有效降低 Java 应用程序的大小(Size)。

2、亲自开发一个Native Service : **SQRS**Service









- 此核心服务命名为SQRService，它做简单的整數平方(Square)运算。其开发要点为：
 1. 核心服务通常在特定的进程(Process)裡执行。
 2. 必须提供 IBinder 接口，让应用程序可以进行跨进程的绑定(Binding)和调用。

3. 因为共享，所以必须确保多线程安全 (Thread-safe)。
4. 以 C++ 類定义，诞生其对象，透过 SM 之协助，将该对象參考值传给 `IServiceManager::addService()` 函數，就加入到 Binder Driver 裡了。

5. 应用程序可透过 SM之协助而远距绑定该核心服务，此时SM会回传IBinder接口给应用程序。
6. 应用程序可透过 IBinder::transact()函數來与核心服务互传资料。

亲自撰写

階段一：撰寫服務

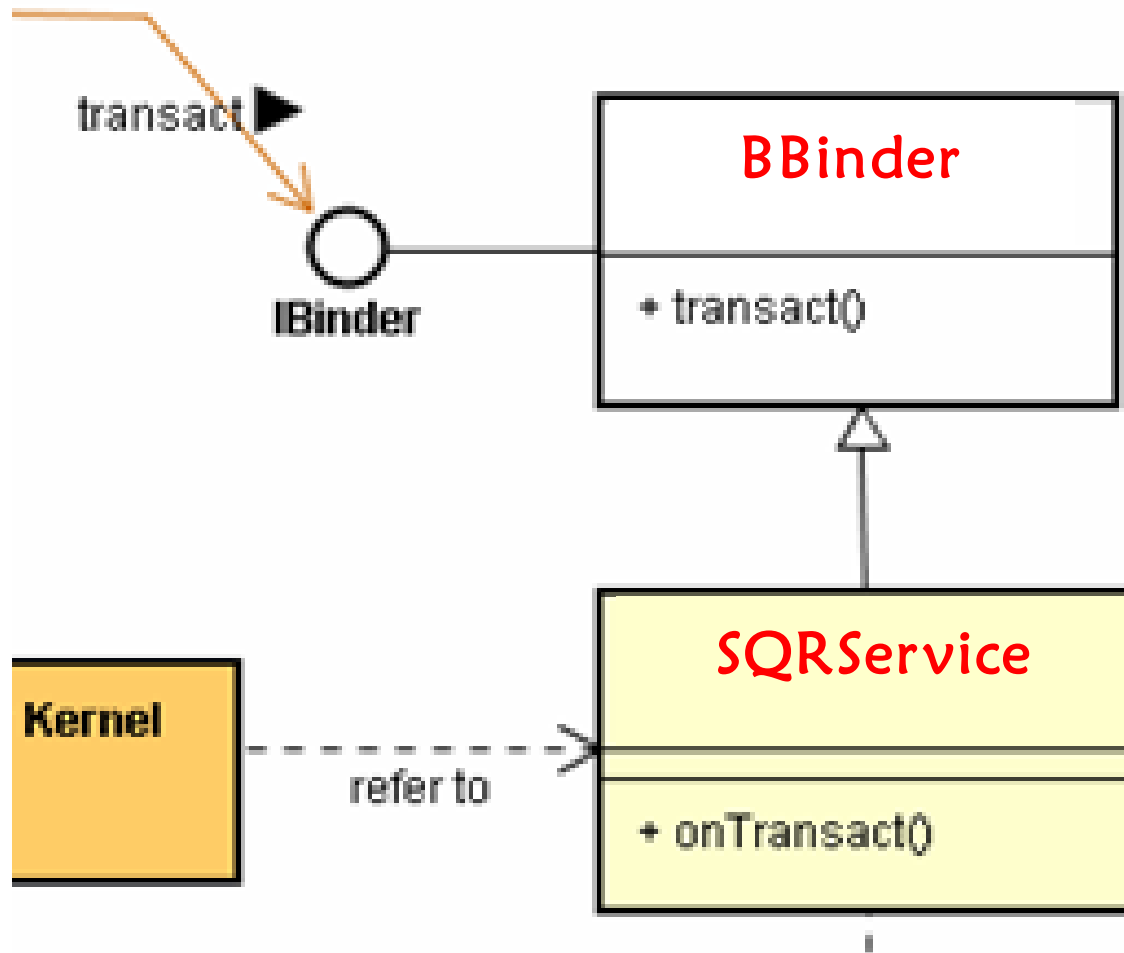
以C++撰写一个SQRService

- **步骤1.1:** 以C++撰写一个SQRService，它的工作是将一个数值(例如5)做开平方的运算，然后传出其计算结果(例如5的平方值为25)。于是，撰写一个C++类别(名称为SQRService)，如下述的程序码：


```
// SQRService.h
#include <stdint.h>
#include <sys/types.h>
#include <utils/Parcel.h>
#ifndef ANDROID_MISOO_SQRSERVICE_H
#define ANDROID_MISOO_SQRSERVICE_H
#include <utils.h>
#include <utils/KeyedVector.h>
#include <ui/SurfaceComposerClient.h>
namespace android {
```

```
class SQRService : public BBinder
{
    public:
        static int instantiate();
        virtual status_t onTransact(uint32_t, const Parcel&,
                                    Parcel*, uint32_t);

        SQRService();
        virtual ~SQRService();
};
};
#endif
```



在Java層，Binder類實現了IBinder接口。

在C++層，BBinder類實現IBinder接口。

```
// SQRService.cpp
#include <utils/IServiceManager.h>
#include <utils/IPCThreadState.h>
#include <utils/RefBase.h>
#include <utils/IInterface.h>
#include <utils/Parcel.h>
#include "SQRService.h"
namespace android {
enum {
    SQUARE = IBinder::FIRST_CALL_TRANSACTION,
};
int SQRService::instantiate(){
    LOGE("SQRService instantiate");
    int r = defaultServiceManager()->addService(
        String16("misoo.sqr"), new SQRService());
    LOGE("SQRService r = %d\n", r);
    return r;
}
```

```
SQRService::SQRService(){ LOGV("SQRService created"); }
SQRService::~SQRService(){ LOGV("SQRService destroyed"); }
status_t SQRService::onTransact(
    uint32_t code, const Parcel& data, Parcel* reply, uint32_t flags) {
    switch(code) {
        case SQUARE: {
            int num = data.readInt32();
            reply->writeInt32(num * num);
            LOGE("onTransact::CREATE_NUM.. n=%d\n", num);
            return NO_ERROR;
        }
        break;
        default:
            LOGE("onTransact::default\n");
            return BBinder::onTransact(code, data, reply, flags);
    }
}; // namespace android
```

- **步骤1.2:** 撰写完成之后，就可以编辑一个 Make 档案，如下：

```
// Android.mk  
LOCAL_PATH:= $(call my-dir)  
include $(CLEAR_VARS)  
LOCAL_SRC_FILES:= \  
SQRService.cpp  
LOCAL_C_INCLUDES := \  
$(JNI_H_INCLUDE)  
LOCAL_SHARED_LIBRARIES := \  
libutils  
LOCAL_PRELINK_MODULE := false  
LOCAL_MODULE := libSQRS01  
include $(BUILD_SHARED_LIBRARY)
```


- 执行此Android.mk 就可进行编译(Compile)和连结(Link)，来产出libSQRS01.so 共享库。

- **步骤1.3:** 使用 C++ 撰写可独立执行的 addserver.cpp 程序。此程序先创建 SQRService 的对象，然后将它 Binder Driver 裡。此 addserver.cpp 的内容如下：

暂时不改写init.rc文档，
以及重新启动设备。

先使用addserver程序
来测试这个核心服务。

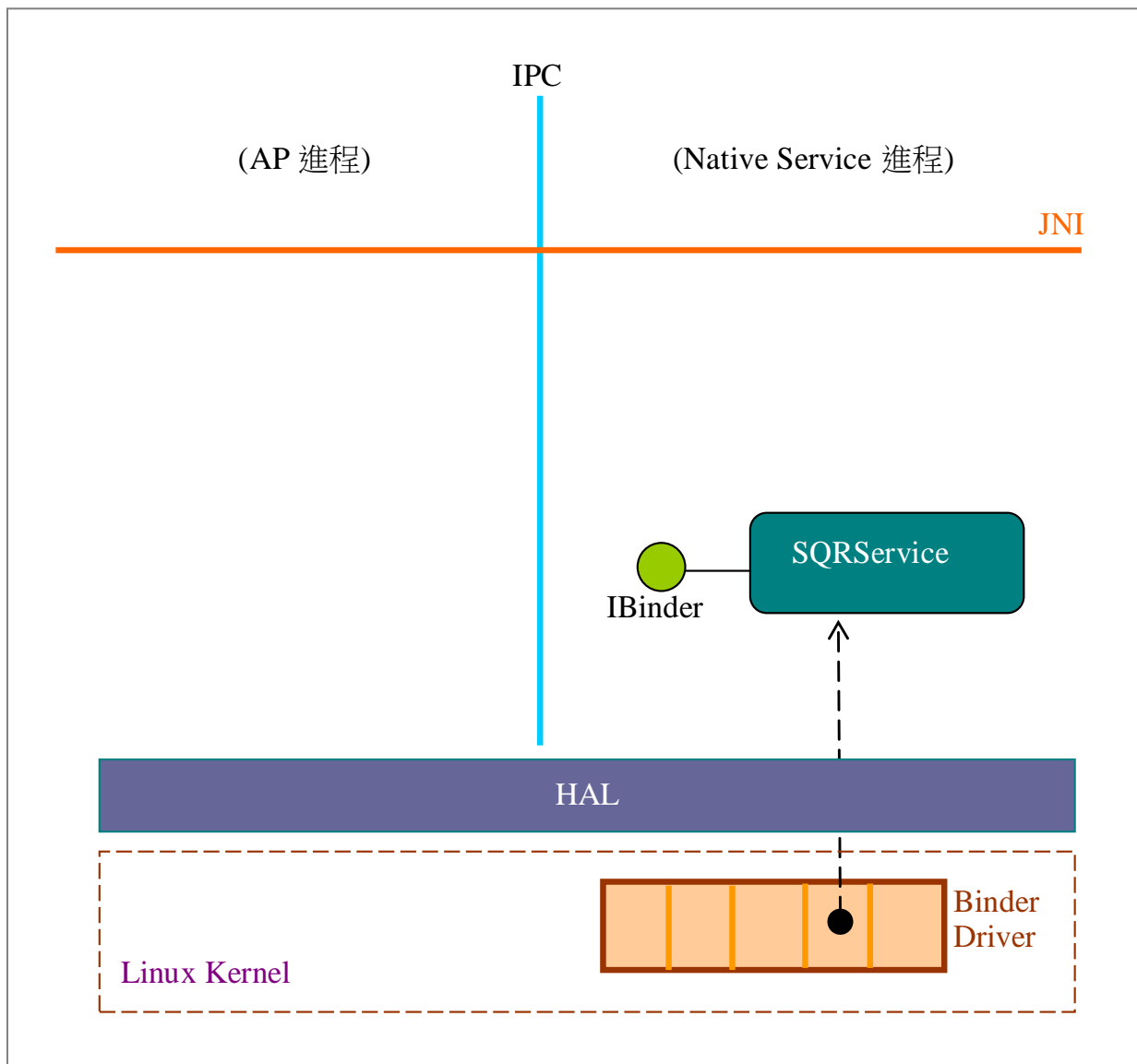
```
// addserver.cpp
#include <sys/types.h>
#include <unistd.h>
#include <grp.h>
#include <utils/IPCThreadState.h>
#include <utils/ProcessState.h>
#include <utils/IServiceManager.h>
#include <utils/Log.h>
#include <private/android_filesystem_config.h>
#include "../libadd/SQRService.h"
//#include <libadd/SQRService.h>
using namespace android;
```

```
int main(int argc, char** argv)
{
    sp<ProcessState> proc(ProcessState::self());

    sp<IServiceManager> sm = defaultServiceManager();
    LOGI("ServiceManager: %p", sm.get());
    SQRService::instantiate();

    ProcessState::self()->startThreadPool();
    IPCThreadState::self()->joinThreadPool();
}
```

- 上述步骤，分别建立了libSQRS01.so 和 addserver 可执行程序。
- 然后将libSQRS01.so 放入Android 仿真器的/system/lib 裡。并且，将addserver 程序放入Android仿真器的/system/bin 裡。
- 执行 addserver 程序，就将SQRService 对象加入Binder Driver 裡。



先不要杀掉addserver进程。





~ Continued ~