

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>

D01\_a

# 认识Android核心服务(a)

By 高煥堂

# 内容

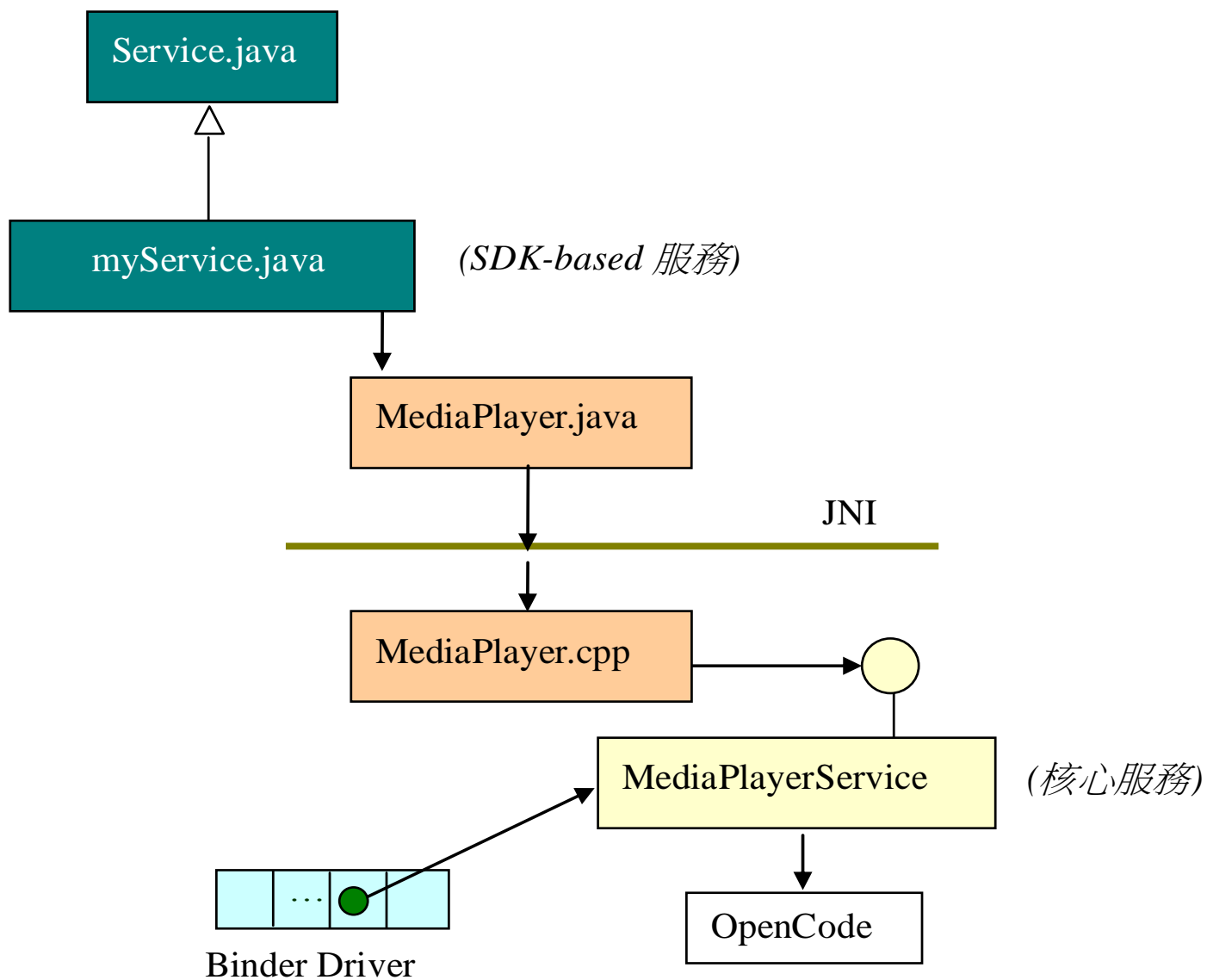
1. 区分核心服务与App服务
2. 回忆App服务
3. 区分两种核心服务：Android Service与Native Service
4. 核心服务的特性
5. 简介几个Android Service
6. 简介几个Native Service
7. 天字第一号的核心服务: ServiceManager
8. 核心服务的共通API：IBinder接口
9. BBinder基类实现IBinder接口

# 1、区分核心服务与App服务

- 在Android平台，上层App的SDK-based Service与底层的核心服务有很大的区别，只是许多人都将它们混在一起了。
- 于此，兹说明Android里有2种服务：
  1. 应用层的应用服务，通称为SDK-based Service(或称为 App服务)。
  2. 系统层的核心服务，通称为Core Service(或称为System Service)。

# App服务

- 至于App服务(即SDK-based Service)则属于App的一部份，是开机完成后，用户加载并开启某App时，才会启动该App里的服务。这种App服务的特征是，它会定义成为Service的一个子类别。
- 如下图的myService子类别，它继承Android应用框架里的Service基类。



- 在Android开机过程中，就会启动数十个核心服务。当开机完成时，全部核心服务也都启动完成了。
- 接着，才能启动App，并执行之。所以，核心服务的启动时间早于App的启动时间。

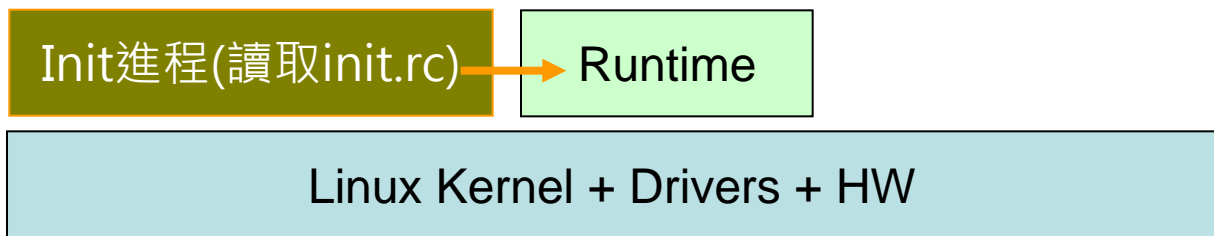


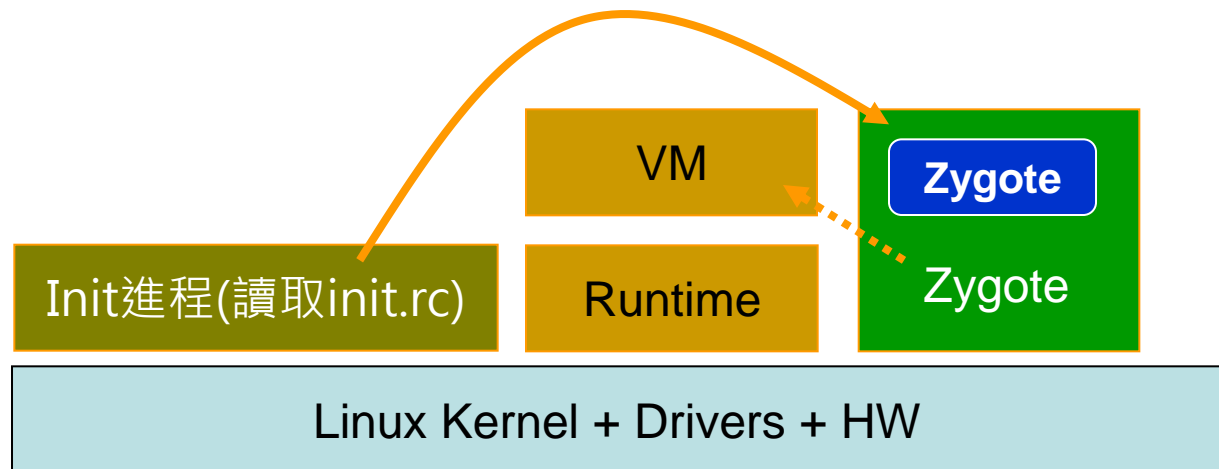
Linux Kernel + Drivers + HW

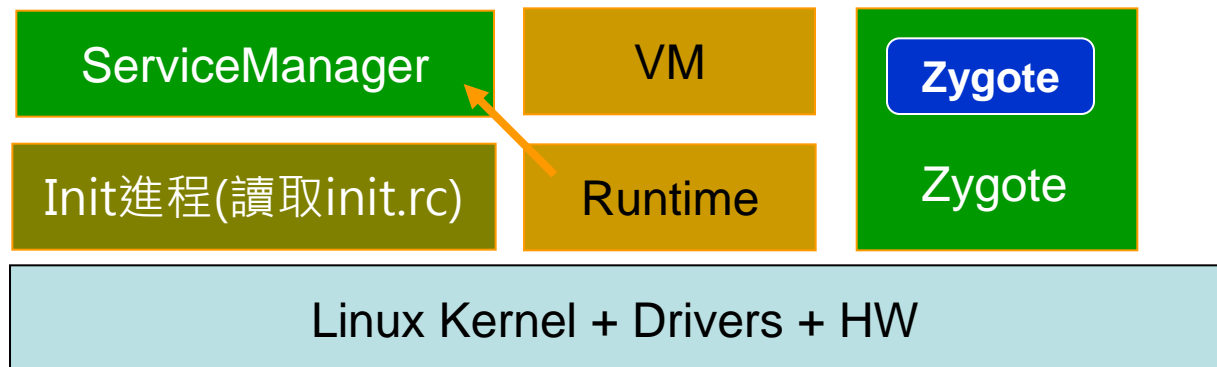
Init進程(讀取init.rc)

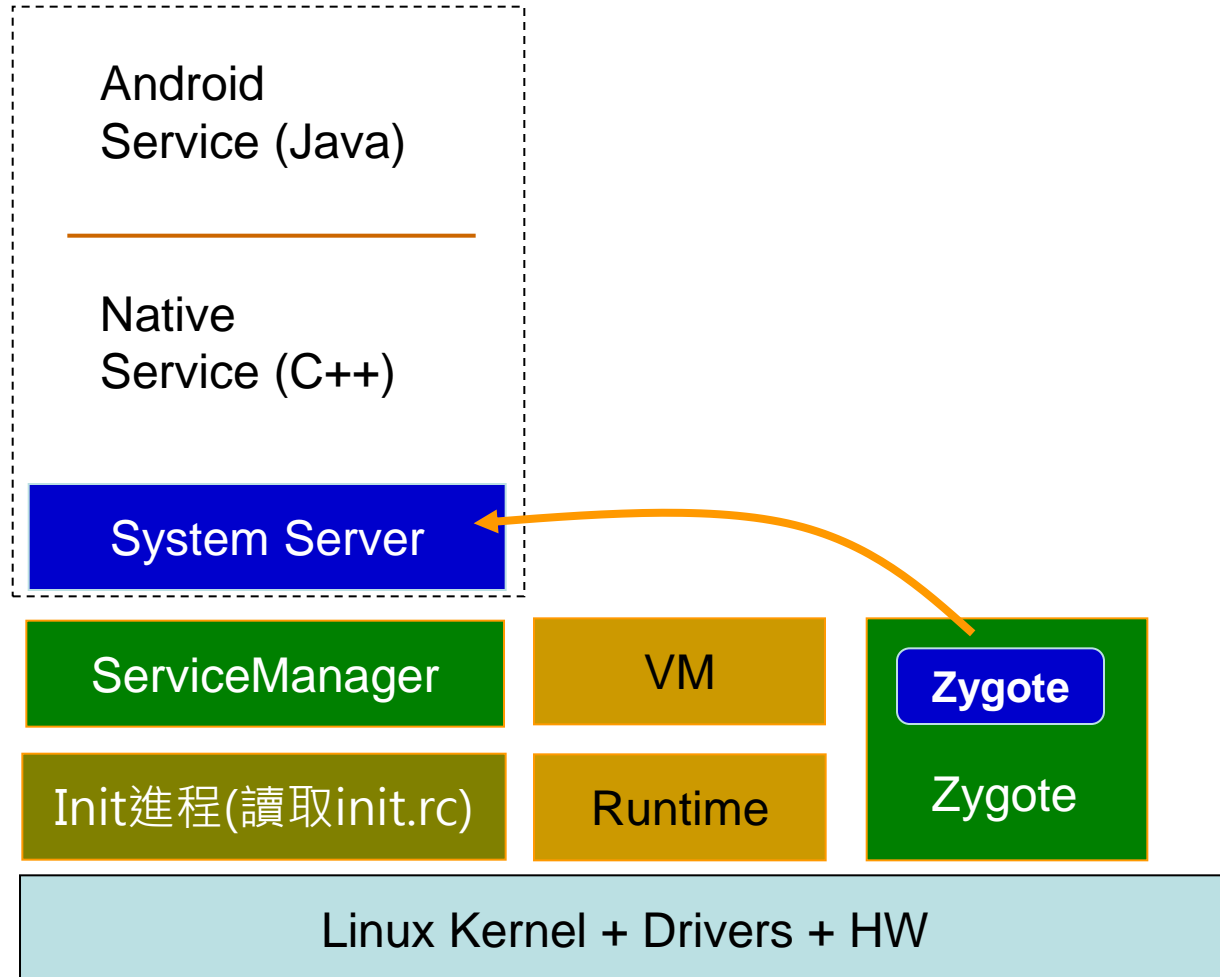
The diagram consists of two rectangular boxes. The bottom box is light blue and contains the text 'Linux Kernel + Drivers + HW'. The top box is yellow and contains the text 'Init進程(讀取init.rc)'. An orange arrow points upwards from the top edge of the blue box to the bottom edge of the yellow box, indicating that the Init process runs on top of the kernel and hardware.

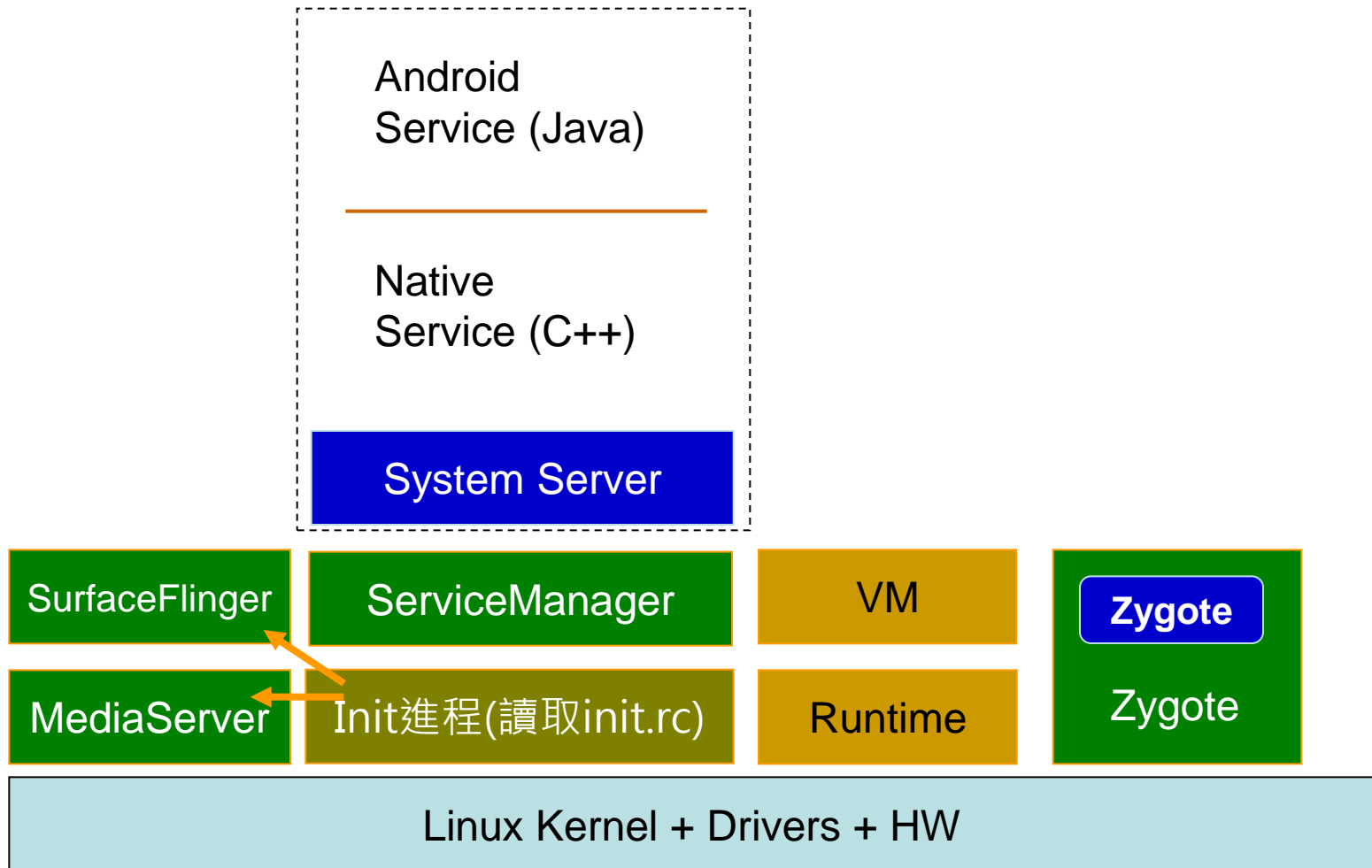
Linux Kernel + Drivers + HW

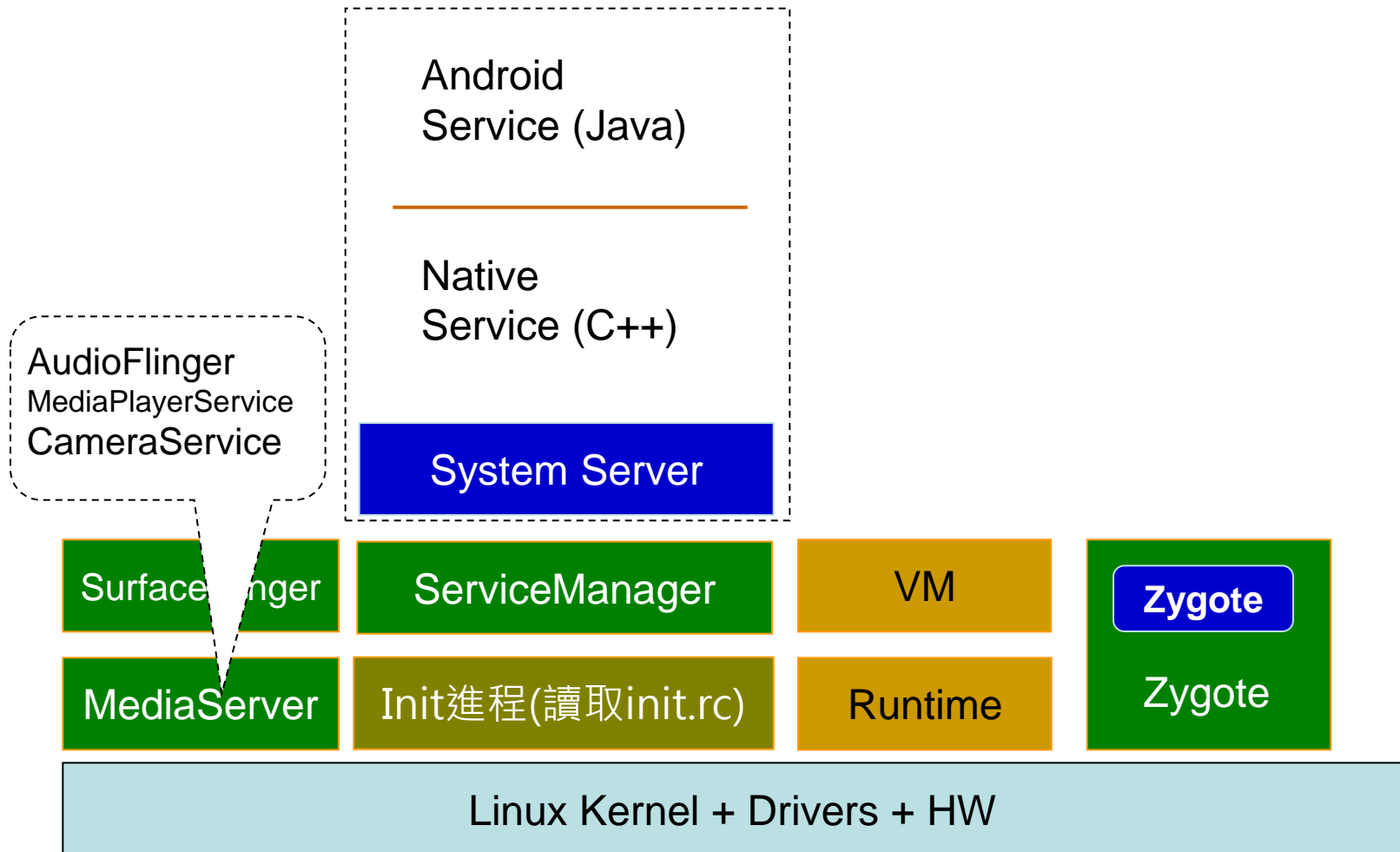




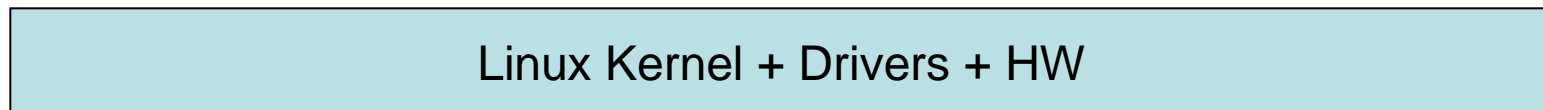
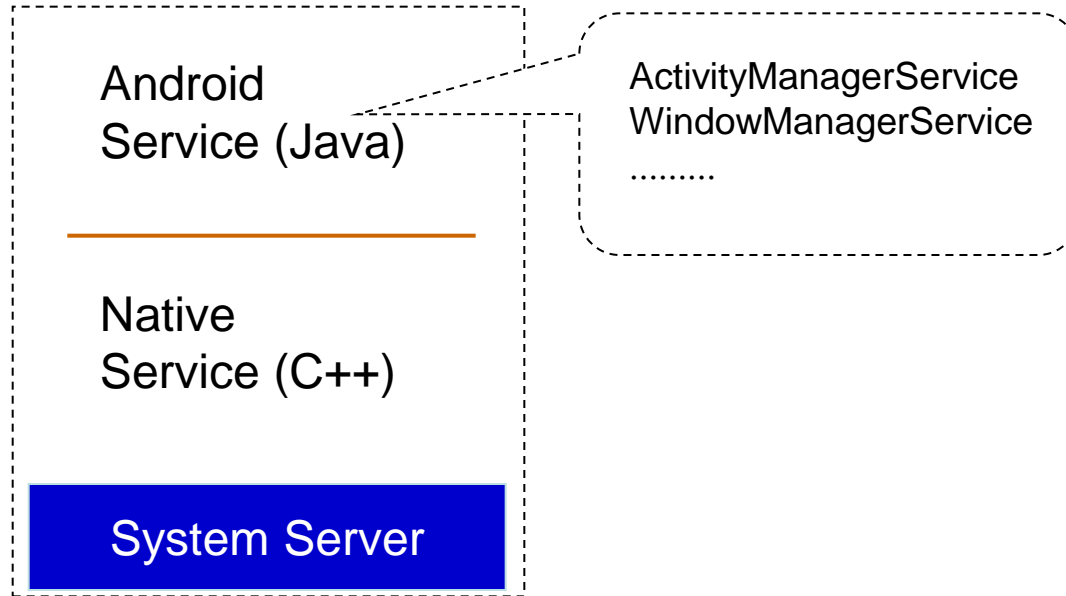




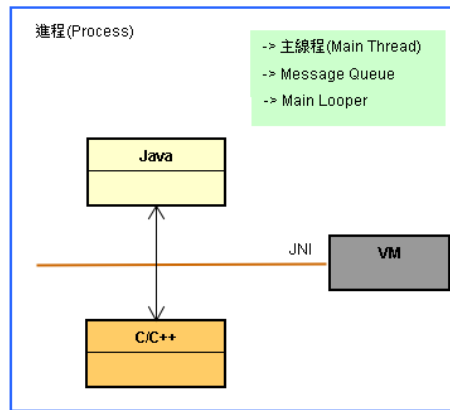








- AMS(ActivityManagerService)启动一个App的时候，就会通过Socket请求Zygote来fork一个进程给这个即将要启动的App。



System Server

SurfaceFlinger

ServiceManager

VM

Fork Service

MediaServer

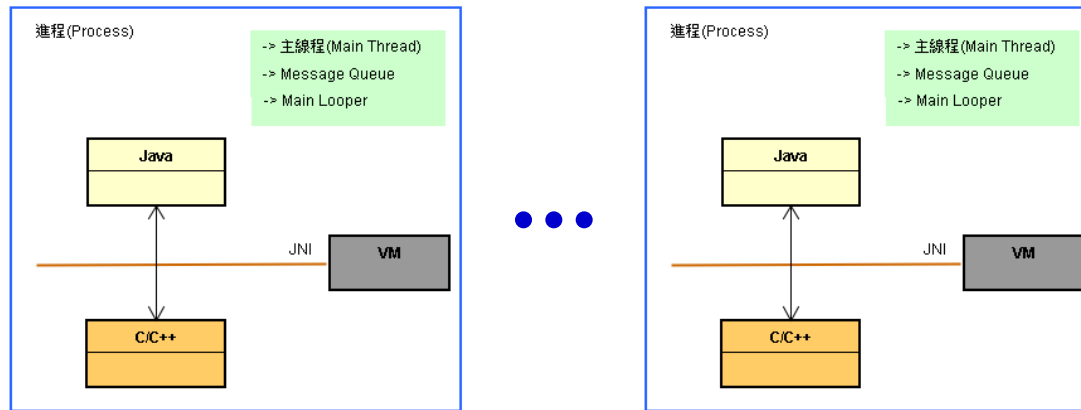
Init進程(讀取init.rc)

Runtime

Zygote

Linux Kernel + Drivers + HW

fork



System Server

fork

SurfaceFlinger

ServiceManager

VM

Zygote

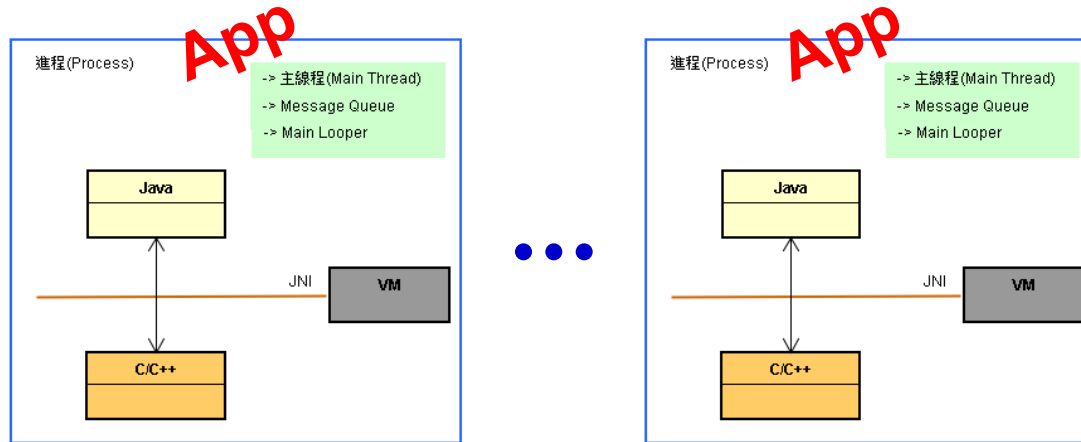
MediaServer

Init進程(讀取init.rc)

Runtime

Zygote

Linux Kernel + Drivers + HW



System Server

SurfaceFlinger

ServiceManager

VM

Zygote

MediaServer

Init進程(讀取init.rc)

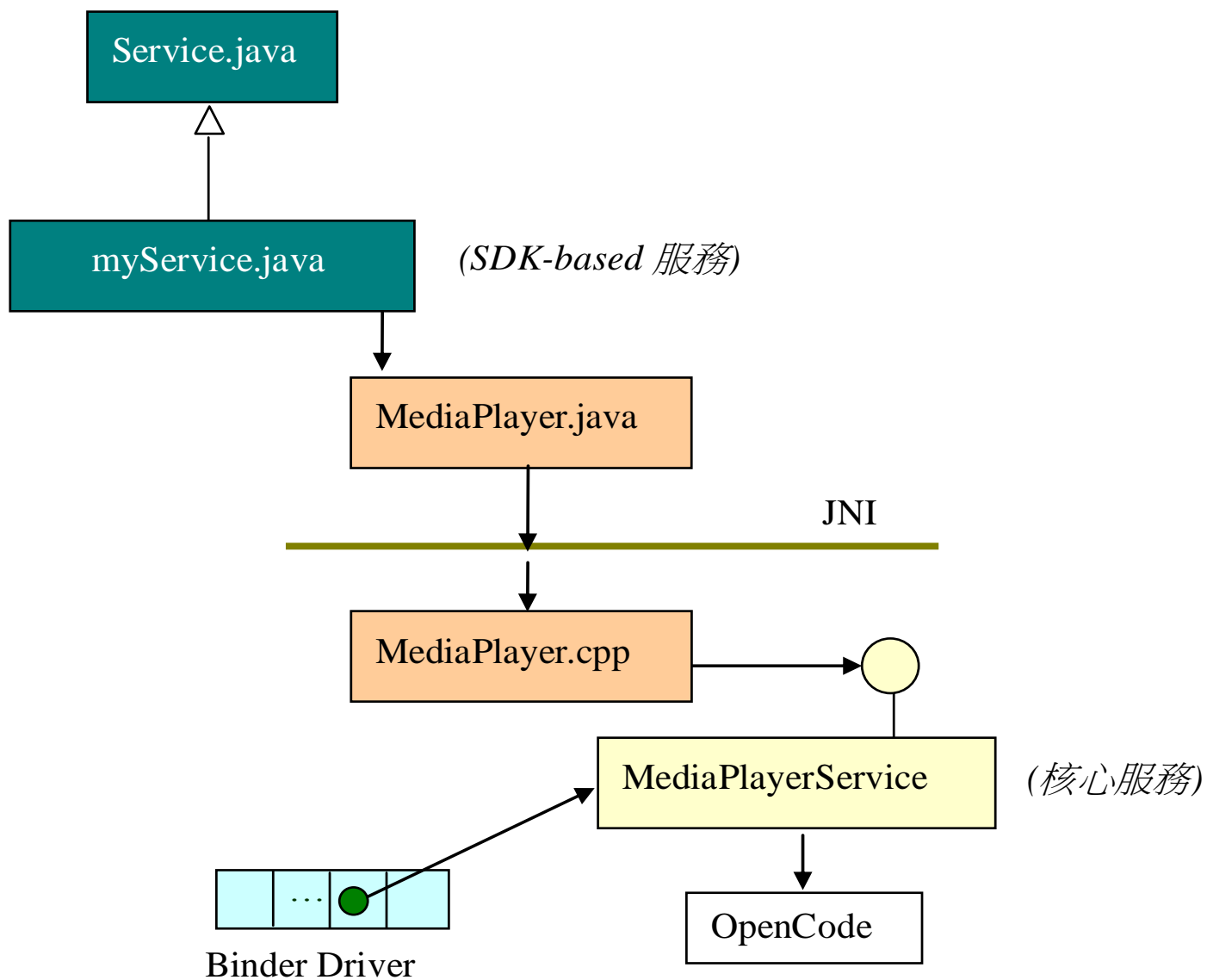
Runtime

Zygote

Linux Kernel + Drivers + HW

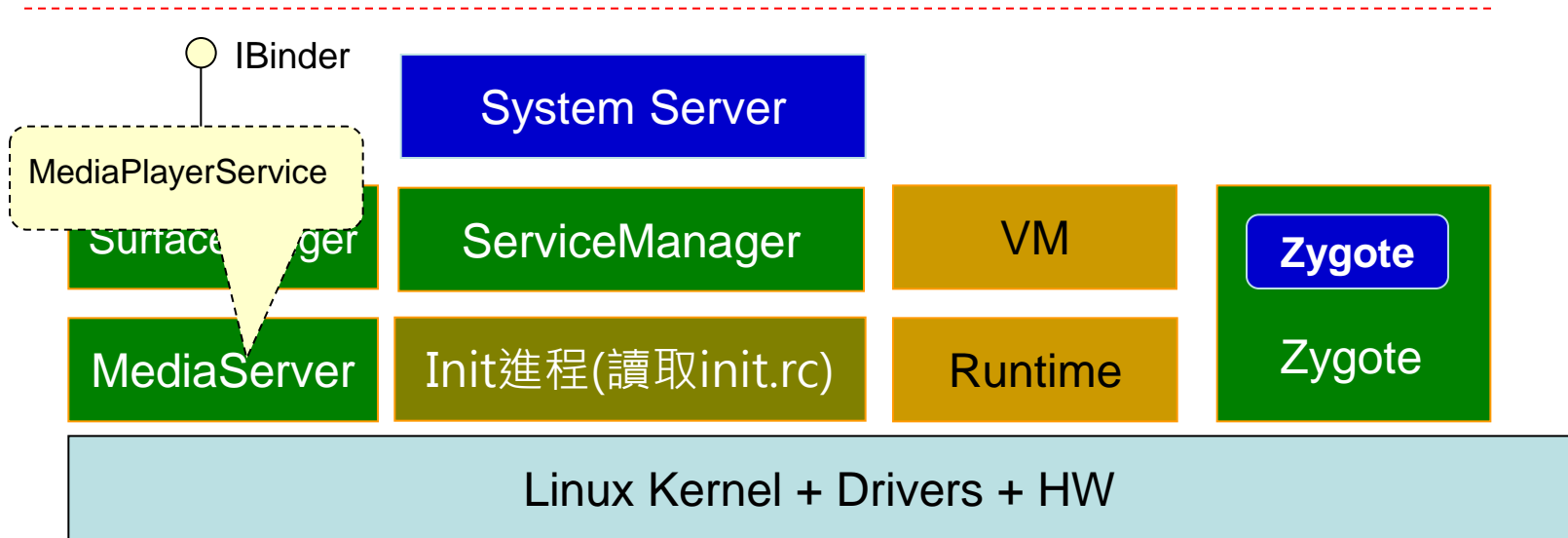
**System**

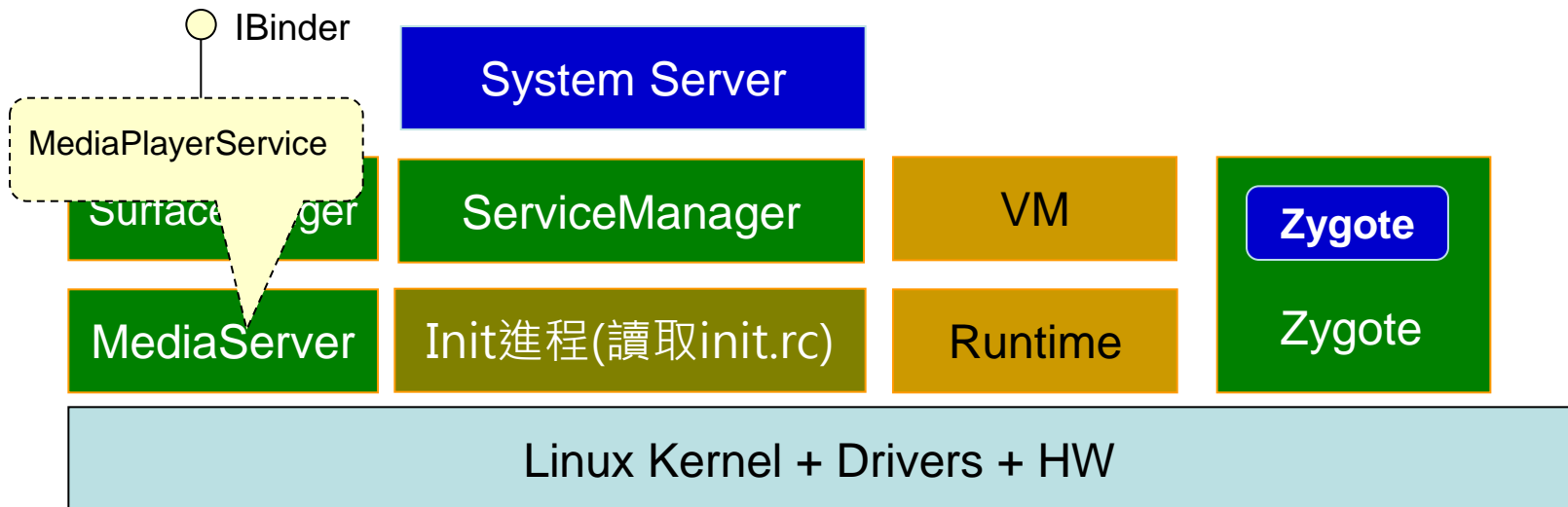
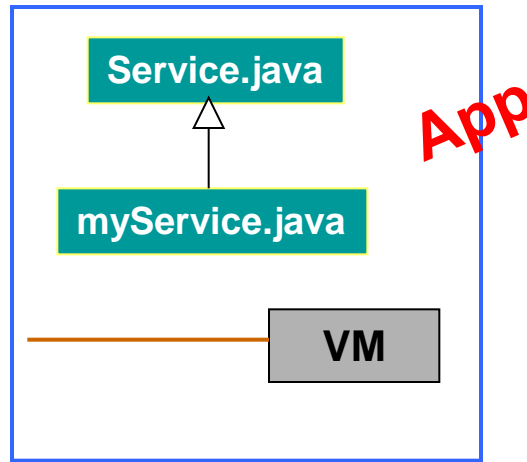
- 也就是说，当App服务启动时，核心服务早已经启动存在了。
- 因此，myService可以透过MediaPlayer和JNI来绑定(Bind)和调用核心服务MediaPlayerService。

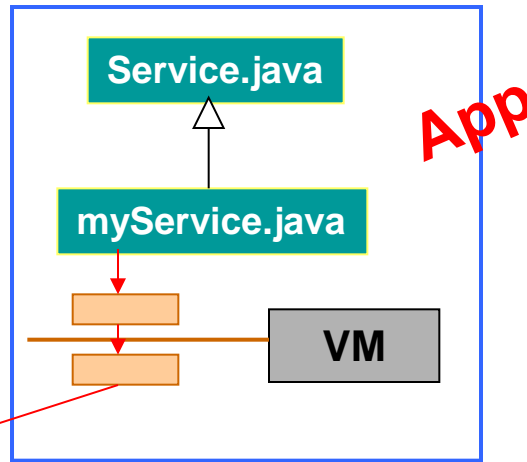


- 其中的myService就是一个App服务；而MediaPlayerService就是一个核心服务。到此，已经初步介绍了核心服务的特性，让你能对分辨核心服务和App服务的微妙区别了。

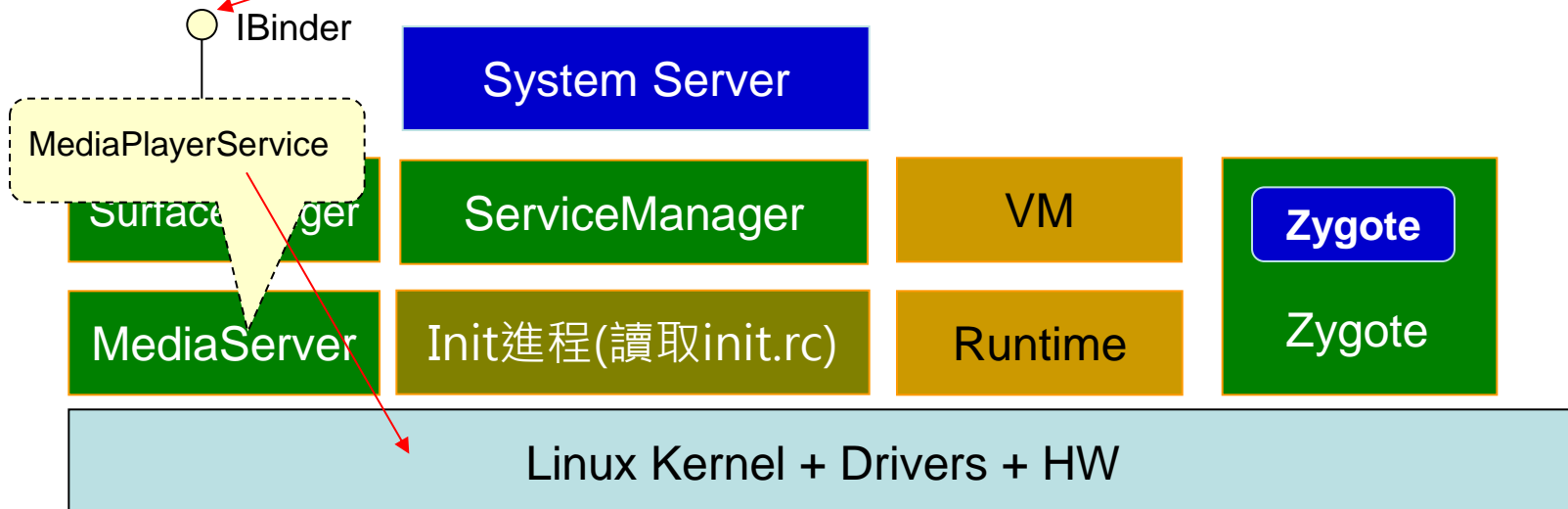








App

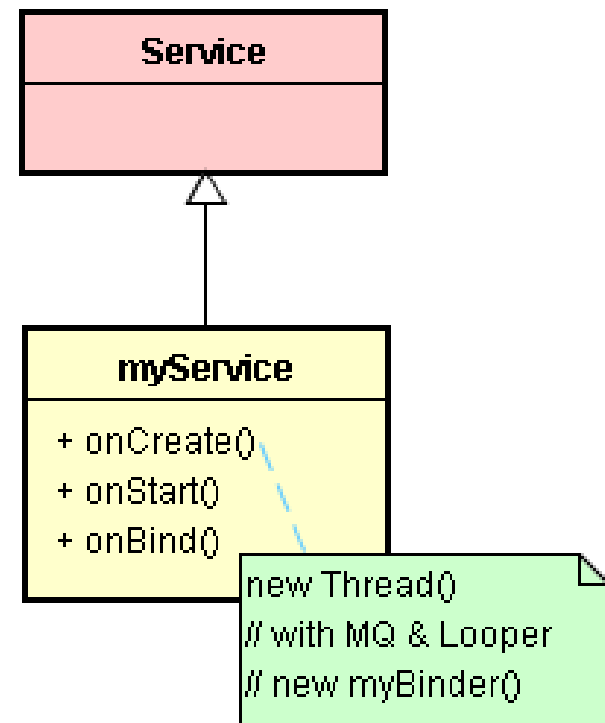
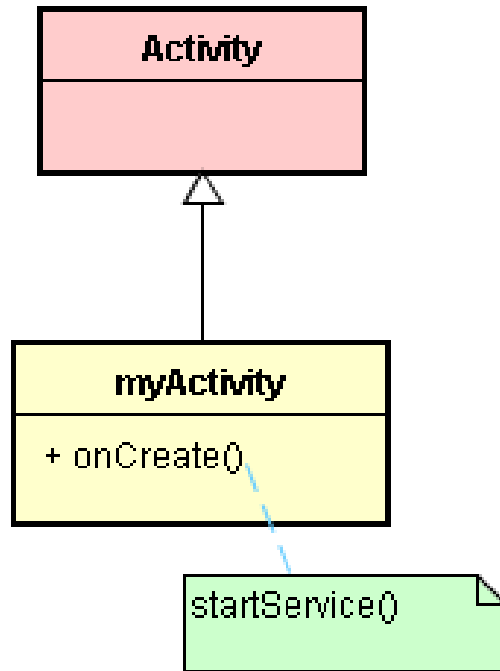


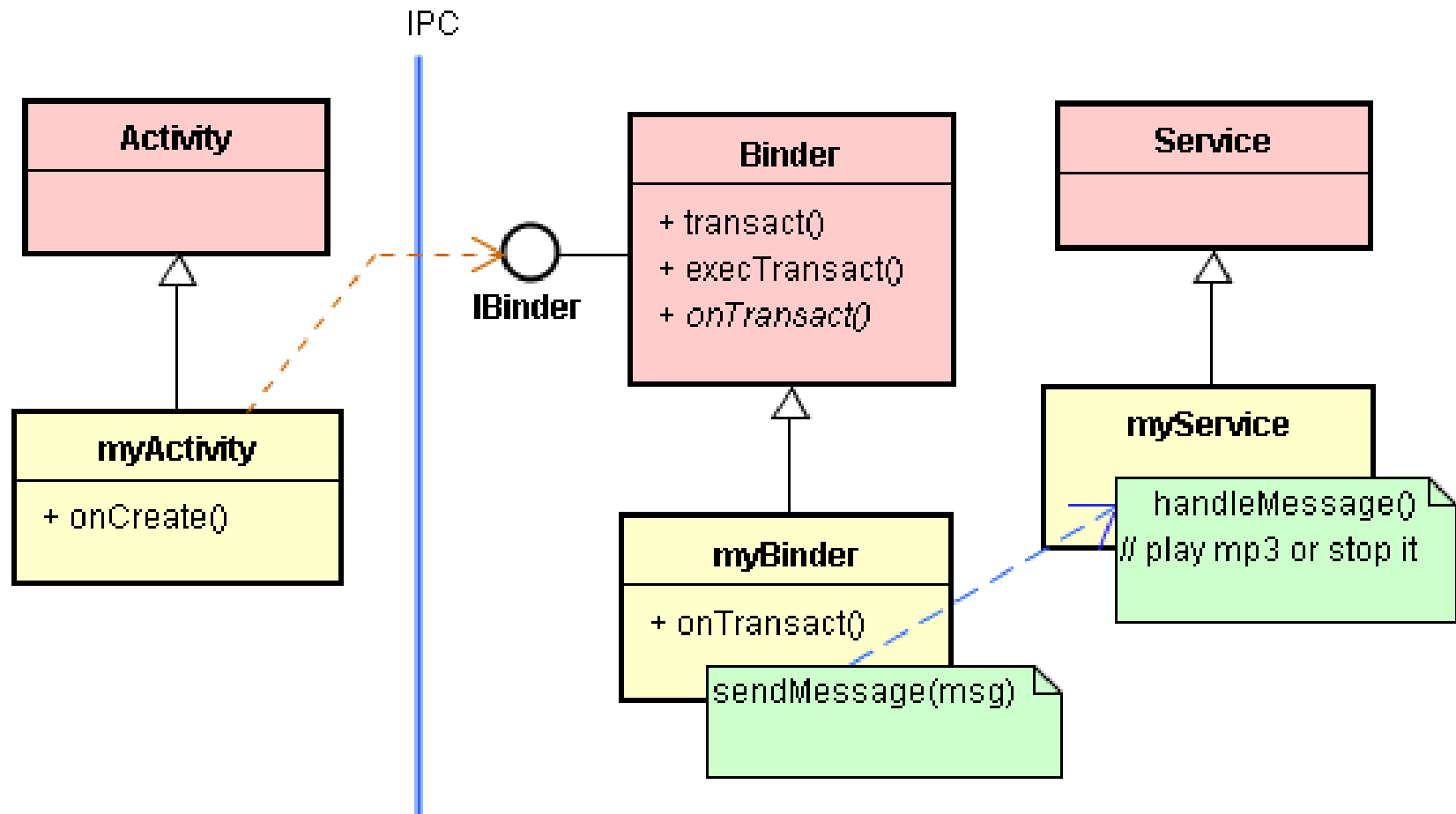
## 2、回忆App服务

- 核心服务是系统层的「前装型」服务
  - 大多以C++类别实现，有些以Java类别实现。
  - 可透过ServiceManager来建立和绑定(Bind)核心服务。
  - 绑定后，可透过IBinder接口去执行其函数。

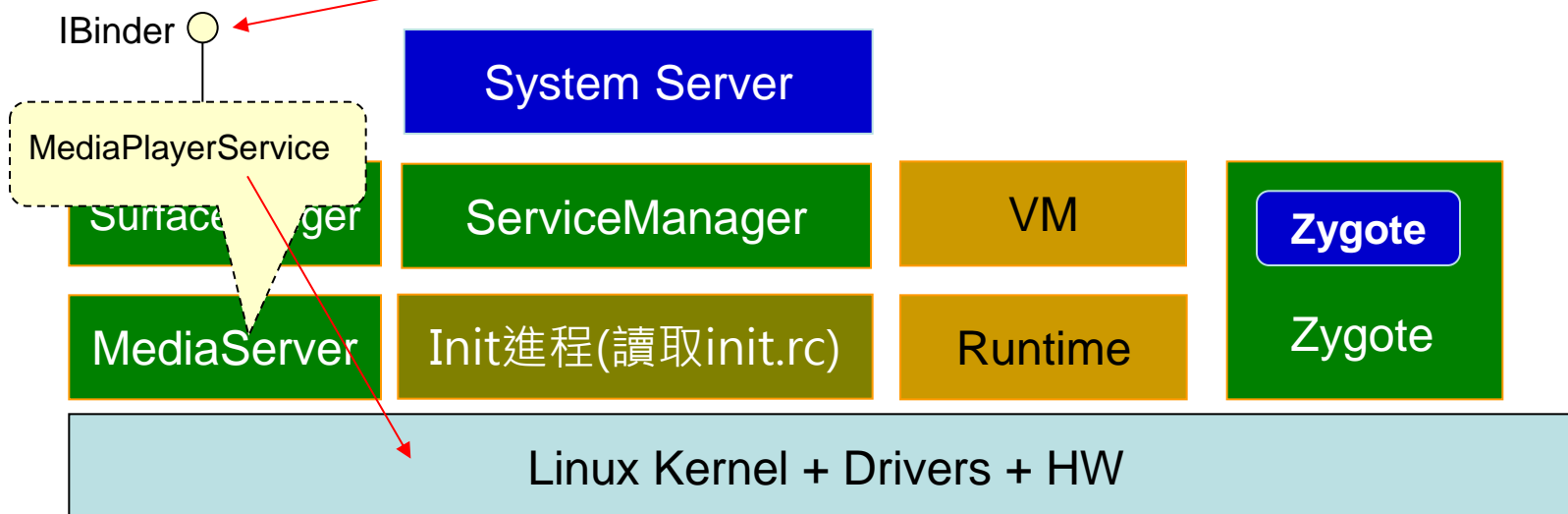
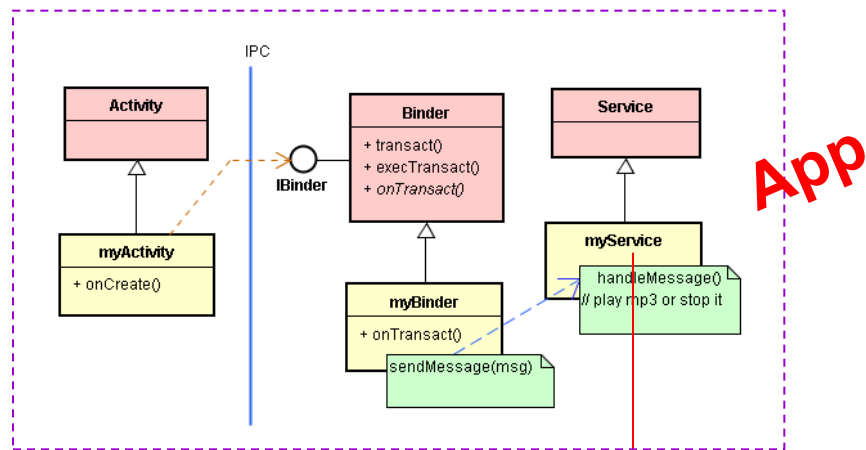
- App服务是应用层的「后装型」服务
  - 以Java层的Service的应用子类别实现。所有的App服务都是由ActivityManagerService核心服务所掌管。
  - 在应用层(如Activity)可调用Android框架里Context类别的startService()和bindService()函数去绑定App服务。
  - 绑定后，可透过IBinder接口去执行其函数。

IPC











**~ Continued ~**