# Android

## Sources:

<https://developer.android.google.cn/>

<https://source.android.google.cn/devices/architecture?hl=zh-cn>

## Makefile skill

### P

#### LOCAL\_PRIVATE\_PLATFORM\_APIS

LOCAL\_PRIVATE\_PLATFORM\_APIS := true

设置后，会使用sdk的hide的api來编译

在Android.mk中如果有

LOCAL\_SDK\_VERSION 这个编译配置，

就会使编译的应用不能访问hide的api,

有时一些系统的class被import后编译时说找不到这个类，就是这个原

因造成的。

LOCAL\_SDK\_VERSION := current 意思是编译时忽略系统隐藏类(@hide)

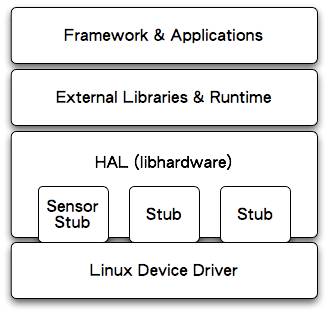
#### LOCAL\_EXPORT\_PACKAGE\_RESOURCES

添加系统级资源

## Conception

### HAL

硬件抽象层是介于android内核kernel和上层之间的抽象出来的一层结构。他是对linux驱动的一个封装，对 上层提供统一接口，上层应用不必知道下层硬件具体怎么实现工作的，它屏蔽了底层的实现细节



1.谷歌搭好了hal的框架，为上层framework打通过jni调用hal提供了统一的api，硬件开发商或者移植人员只需要按照框架开发即可，无需话费精力在与上层的交互上的实现上，将精力放在hal层本身的实现上即可。

2.从商业角度，许多硬件厂商不愿意将自己硬件相关一些核心的东西开源出去，假如将对自己硬件的驱动程序全部放入内核空间驱动程序实现，那么必须遵循GPL协议，是必需开源的。有了HAL层之后，他们可以把一些核心的算法之类的东西的实现放在HAL层，而hal层位于用户空间，不属于linux内核，和android源码一样遵循的是appache协议，这个是可以开源或者不开的。