Android OS  
Overview

Table of Contents

[L’OS Android 1](#_Toc50365283)

[Activity 1](#_Toc50365284)

[Qu’est-ce que c’est ? 1](#_Toc50365285)

[Quand les utiliser ? 1](#_Toc50365286)

[Comment les utiliser ? 1](#_Toc50365287)

[Fragment 1](#_Toc50365288)

[Qu’est-ce que c’est ? 1](#_Toc50365289)

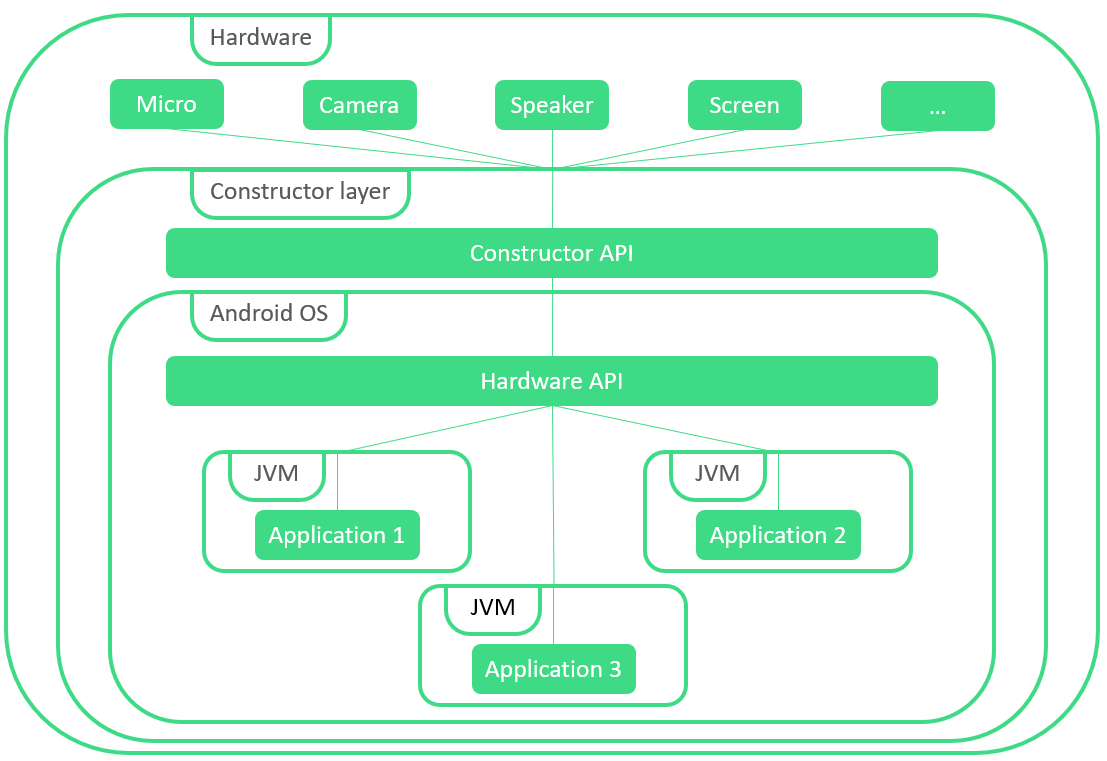
[Quand les utiliser ? 2](#_Toc50365290)

[Comment les utiliser ? 2](#_Toc50365291)

[Recycler View 2](#_Toc50365292)

# L’OS Android

L’OS d’Android est basé sur une technologie de conteneurisation similaire à system D[[1]](#footnote-1). L’OS Android c’est donc « juste » un gestionnaire de container qui est spécialisé dans la gestion de container destinés à encapsuler des JVM[[2]](#footnote-2). Avant de commencer le développement Android, il est important de bien comprendre comment une application interagit avec son système d’exploitation et son hardware. Voilà un petit schéma pour vulgariser ces interactions



On pourra observer sur ce schéma les différentes couches d’un smartphone Android :

* La couche Hardware : théoriquement on ne devrait jamais avoir à toucher cette couche car c’est la couche qui contient tous les composants hardware. On ne veut pas avoir à dessouder l’écran pour changer la couleur d’un pixel.
* La couche Constructeur : Puisque Android est un OS open source, il peut être embarqué sur différentes plateformes par différents constructeurs (Samsung, Huawei, Google, Xiaomi, etc.). Ainsi chaque constructeur utilise ses propres composants hardware et sa propre architecture de composants. Il met alors en place une API qui permet à l’OS Android de pouvoir communiquer avec le hardware. Ainsi chaque téléphone fonctionne différemment en fonction du constructeur qui le fait. Cette couche ne devrait pas impacter nos développements, sauf cas exceptionnel pour certaines features particulières.
* La couche Android OS : C’est là où toute la magie opère. L’OS Android dispose d’une API qui permet aux containers de JVM de « discuter avec le hardware. On voit sur le schéma que l’OS Android démarre une JVM pour chaque application. Cela permet d’avoir des fils d’exécutions bien séparés pour chaque application. Ainsi la gestion de la mémoire et d’erreur est propre à chaque container et donc chaque application.

# Les applications

Une application Android est donc une image système d’un logiciel écrit en Java généré par le SDK d’Android

# Activity

## Qu’est-ce que c’est ?

## Quand les utiliser ?

## Comment les utiliser ?

# Fragment

## Qu’est-ce que c’est ?

Le Fragment c’est une entité qui permet de gérer un ensemble de composant graphique

## Quand les utiliser ?

## Comment les utiliser ?

# Recycler View

1. System D, c’est la technologie de conteneurisation qu’utilise linux pour démarrer son système d’exploitation. [↑](#footnote-ref-1)
2. JVM : Java Virtual Machine. Environnement virtuel autonome dédié à interpréter le code Java. [↑](#footnote-ref-2)