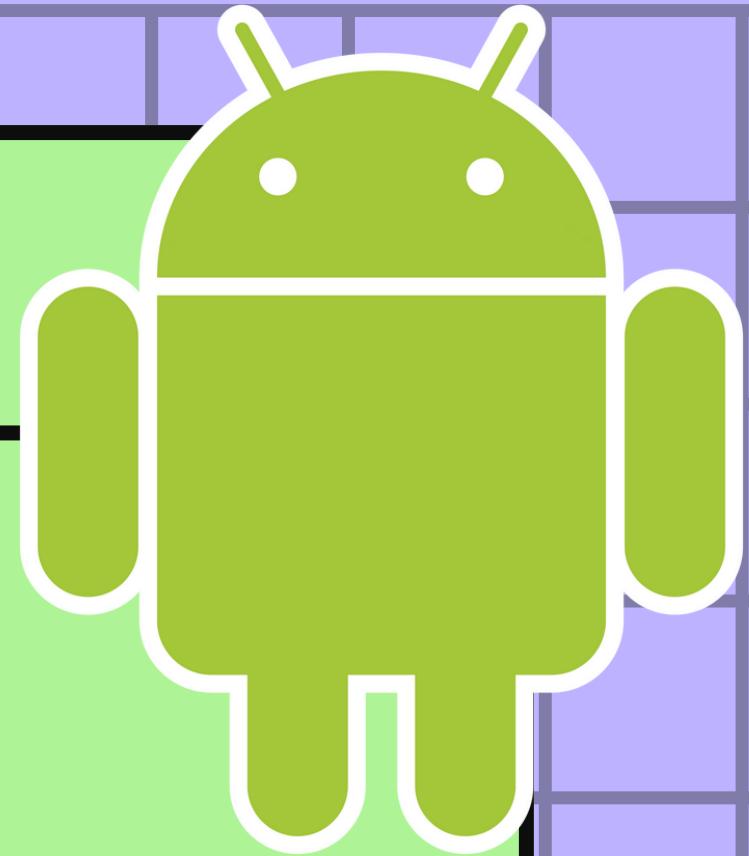




SISTEMAS OPERATIVOS EN
DISPOSITIVOS MÓVILES

ANDROID

ASOP



PRES EN TAOg POA:

CRISTOPHER JUAREZ

y

PABLO CONSTANTINO

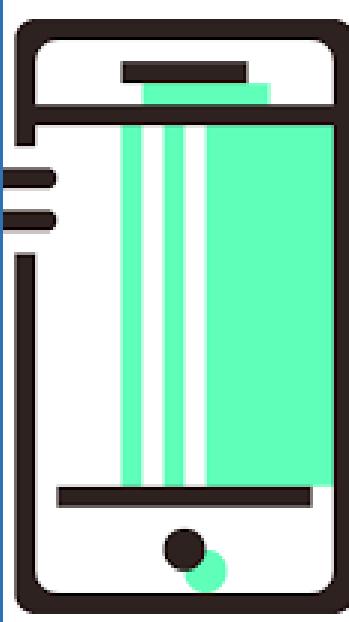


¿QUE ES UN SISTEMA OPERATIVO?

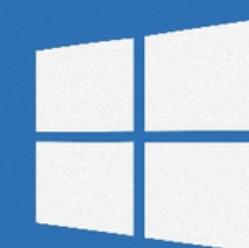
El sistema operativo es el principal programa que se ejecuta en toda computadora de propósito general.
**Gunnar Wolf FUNDAMENTOS DE SISTEMAS
OPERATIVOS**

Un sistema operativo es un programa que controla la ejecución de los programas de aplicación y que actúa como interfaz entre el usuario de un computador y el hardware de la misma. Dr. Vladimir Villarreal **SISTEMAS OPERATIVOS**

SISTEMA OPERATIVO DE DISPOSITIVO MOVIL



X □ -



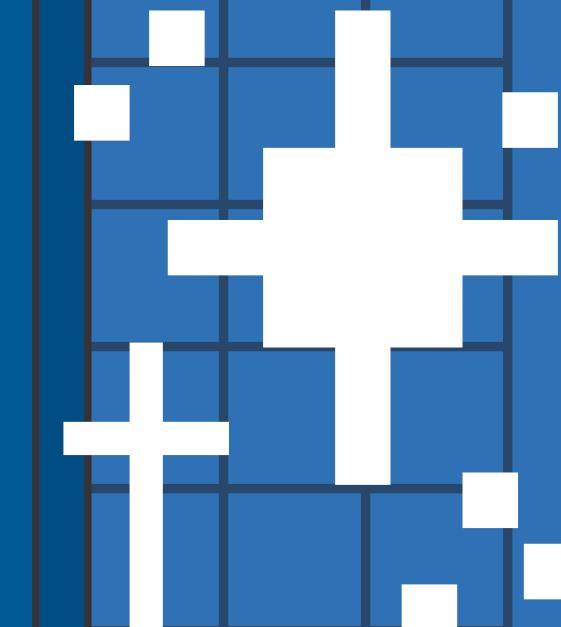
<

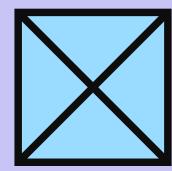
>

Un sistema operativo móvil permite que el software de aplicación funcione en los dispositivos móviles y tabletas.

Un sistema operativo móvil es responsable de administrar los recursos de hardware del dispositivo, como la CPU, la memoria y los dispositivos de entrada y salida, y proporcionar una plataforma para que las aplicaciones se ejecuten en el dispositivo. La empresa fabricante elige el sistema operativo del dispositivo por lo tanto es complicado cambiarlo.

Los sistemas operativos móviles más populares hoy en día son Android de Google y iOS de Apple.





UN POCO DE HISTORIA

symbian
OS
NOKIA
Connecting People

SYMBIAN



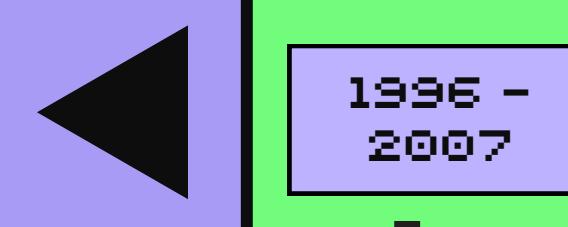
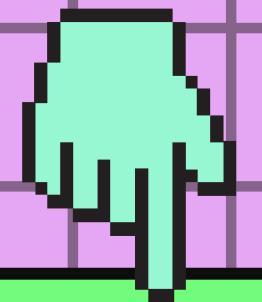
WINDOWS
MOBILE



iOS



UBUNTU
TOUCH



1996 -
2007

1997 -
2012

1999 -
2022

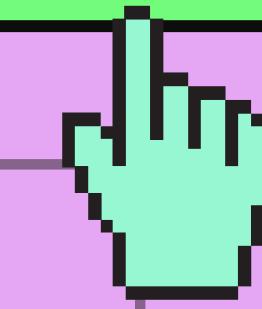
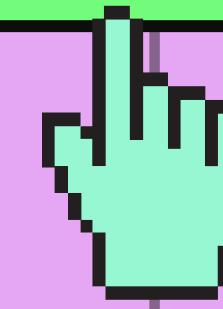
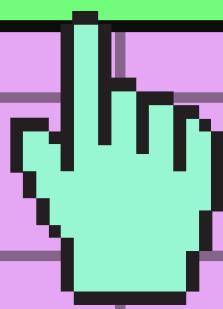
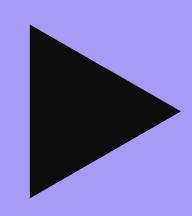
2000 -
2017

2005 -
2011

2007 -
ACTUALIDAD

2008 -
ACTUALIDAD

2010



PALM OS

palm OS 5™

BLACKBERRY OS

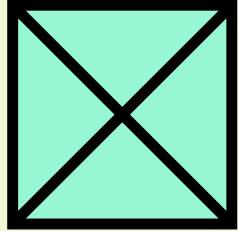


MAEMO



ANDROID



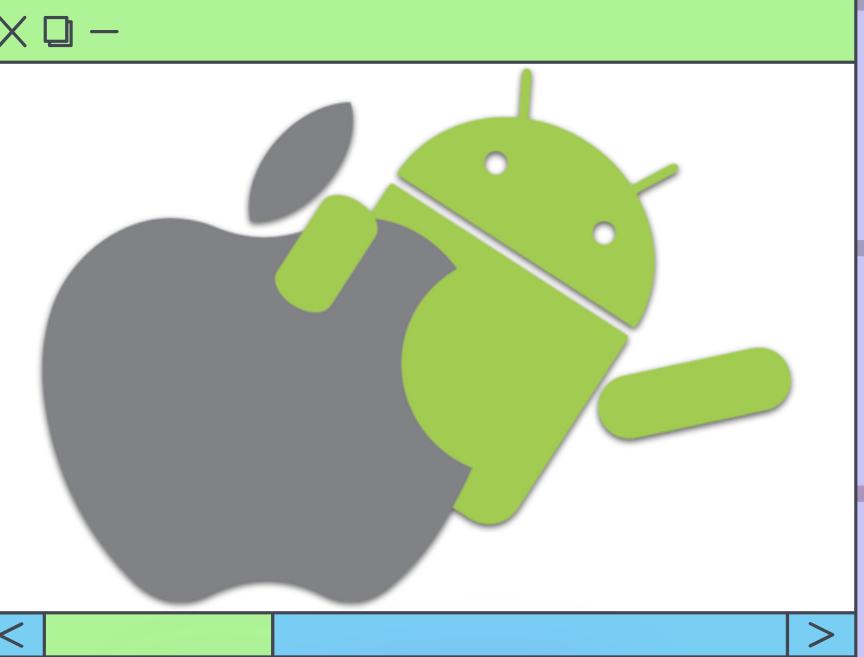


S.O. DESKTOP VS S.O. DISPOSITIVO MOVIL



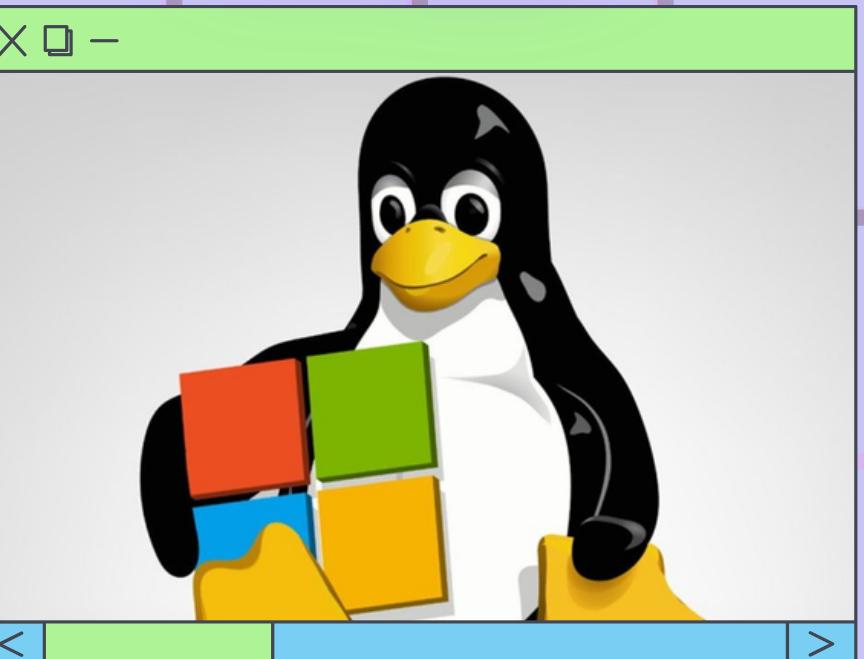
S.O MOVIL

- Interfaces más simplificadas y amigables para el usuario.
- Optimizado para trabajar bajo requisitos mínimos de potencia.
- Suelen ejecutar aplicaciones diseñadas específicamente para ellos.
- Tienden a priorizar el almacenamiento en la nube y el acceso a datos en línea.
- Están diseñados para ser más eficientes y consumir menos energía que los S.O. de las computadoras.



S.O DESKTOP

- Funciona a través de muchos dispositivos de entrada, como mouse, teclado, etc.
- Interfaces más complejas que pueden ser personalizadas por el usuario.
- Ejecutan una amplia variedad de aplicaciones diseñadas para diferentes sistemas operativos.
- Suelen tener más almacenamiento local.
- Utiliza discos duros/ unidades flash para almacenar datos e información.
- Requiere una buena cantidad de memoria para operar.





Android es un sistema operativo abierto, multi-tarea creado para ser independiente de cualquier tipo de arquitectura de hardware en los dispositivos móviles.

Este permite programar aplicaciones en una variación de Java llamada Dalvik. Android proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono (como el GPS, las llamadas, la agenda, etc.) de una forma sencilla en un lenguaje sumamente popular como lo es Java.

Esta característica hace que sea tan popular y atractivo para los fabricantes y desarrolladores.

¿QUÉ ES ANDROID?

android



ARQUITECTURA ANDROID

Es un modelo en capas que describe cómo funciona el sistema operativo Android en un nivel alto de abstracción. La arquitectura de Android está dividida en cuatro capas principales:

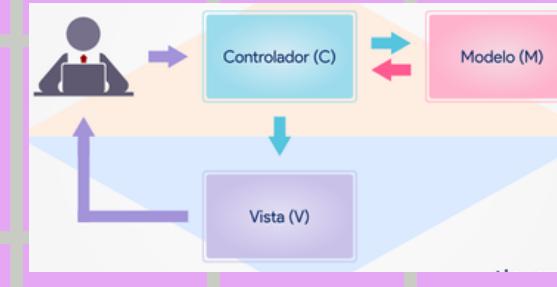
Capa de aplicaciones

- Es donde los usuarios interactúan con las aplicaciones. Ej. redes sociales, juegos, media



Capa de Framework

- Conjunto de servicios para que las aplicaciones de usuario puedan utilizarlos. (gestión de recursos, paquetes, notificaciones).

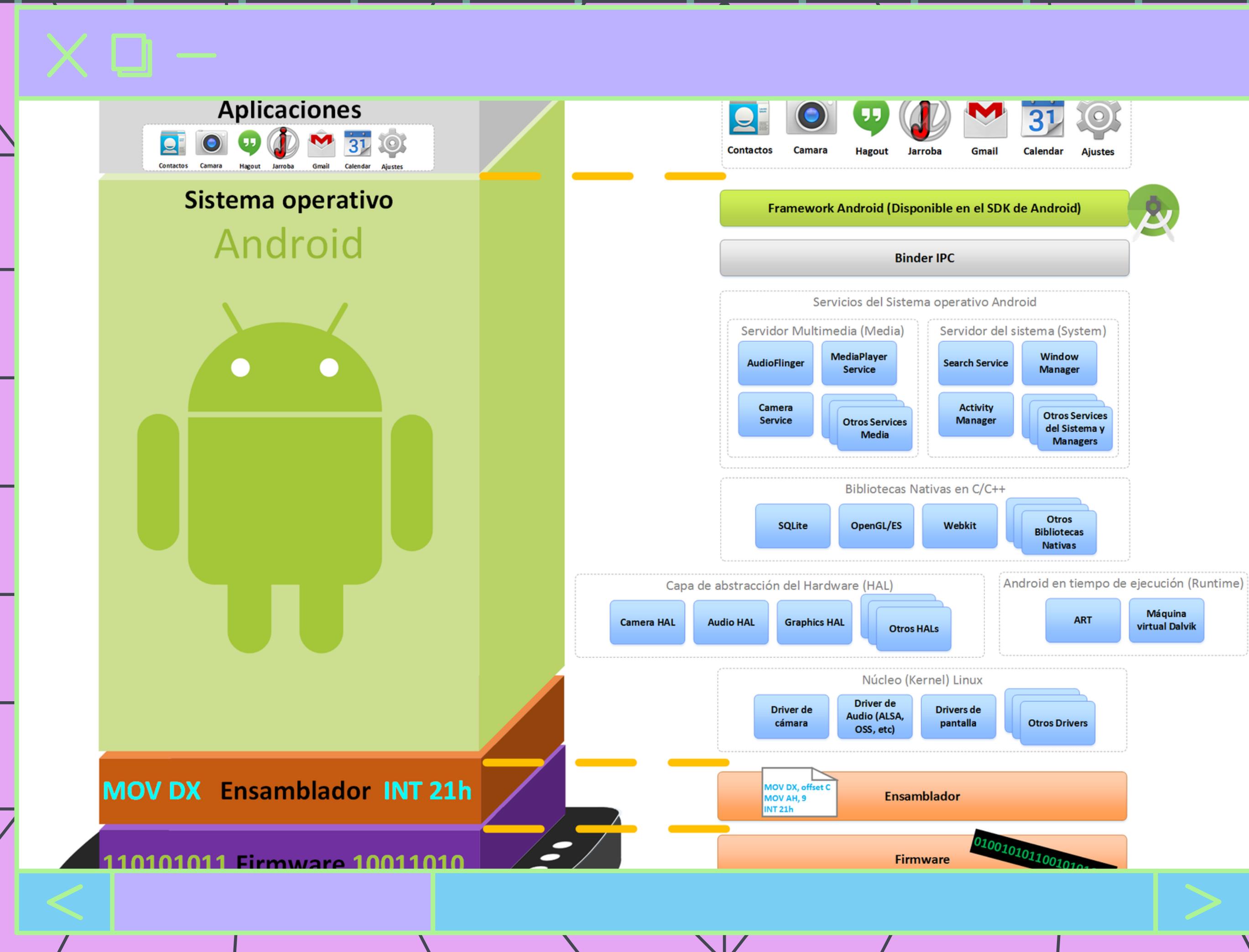


Capa de Biblioteca de Android

- Proporciona un conjunto de bibliotecas C/C++ para dar funciones y servicios a las aplicaciones del usuario

Capa de Kernel de Linux

- Proporciona los controladores de hardware para los componentes del dispositivo, como la pantalla, la cámara, la red, memoria y la CPU.



RELACION ANDROID LINUX



Usualmente conocemos a las distribuciones de Linux como "**Linux**", pero, el término correcto cuando hablas de Ubuntu, Debian etc;es **GNU/Linux**.

Para que dichas distribuciones funcionen necesitan no sólo el **kernel de Linux**, que es la base del sistema operativo, sino el conjunto de software y aplicaciones **GNU**.

Tanto las distribuciones **GNU/Linux** como **Android** usan el **Kernel Linux** como base, pero esto es lo único que tienen en común.

Android está desarrollado con las librerías de **Dalvik** como segunda capa y luego un **framework** que trabaja con **Java**. Mientras que las distribuciones de Linux antes mencionadas incluyen librerías en **C** de **GNU** en vez de la de **Dalvik**.



Dalvik es la máquina virtual usada en los dispositivos Android.

Android junto con Dalvik son los encargados de trabajar con el código fuente Java y convertirlo en código nativo para el Kernel de linux que la plataforma Android soporta.

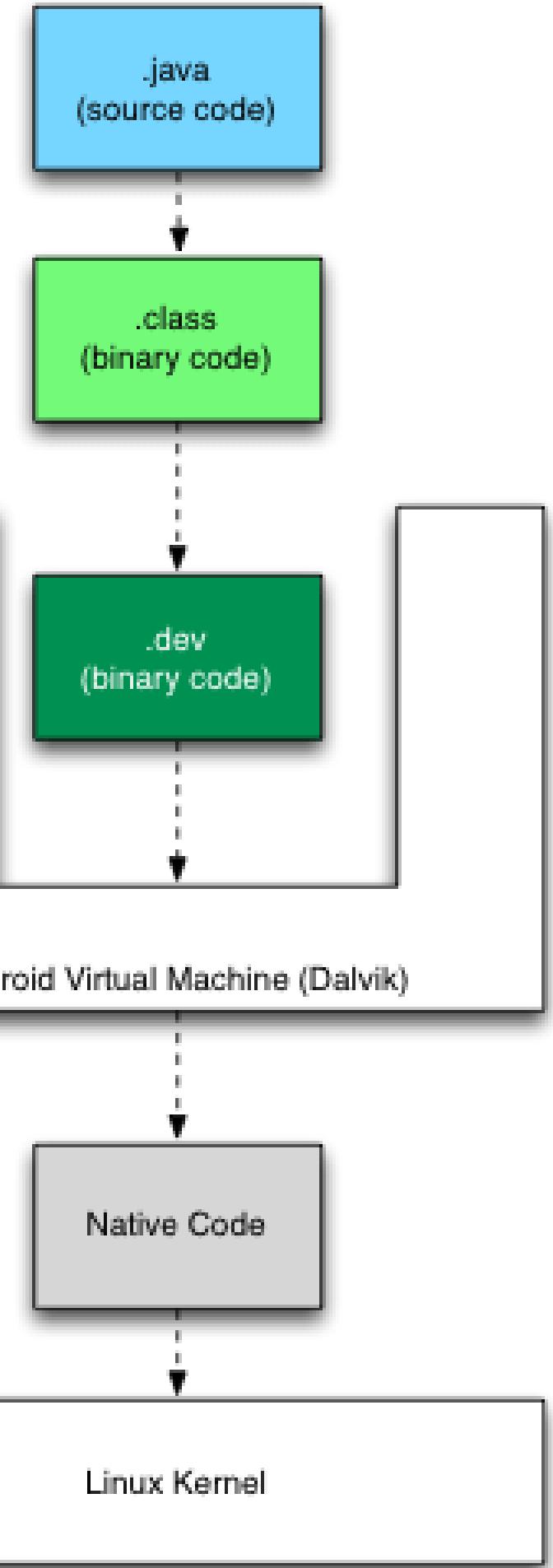
En Android todos los ficheros .class de Java son pasados por un proceso adicional de optimización ya que como sabemos estamos en una plataforma móvil y los recursos son limitados. En dicho proceso todos los ficheros .class son agrupados en un único fichero .dex (Dalvik Executable).

La maquina virtual Dalvik utiliza un modelo de compilación (JIT), que traduce el código de la aplicación en tiempo de ejecución, lo que permite que las aplicaciones sean más rápidas y utilicen menos energía.

Cada aplicación en Android corre en su propio proceso con su propia instancia de la Maquina Virtual.

DALVIK

DALVIK



ART (ANDROID RUNTIME)



ART o Android Runtime es la máquina virtual de Android a partir de la versión 5.0. ART sustituyó a DALVIK por su modelo de compilación anticipada AOT (Ahead-Of-Time). La cual crea un archivo de compilación tras la instalación de una aplicación. De este modo, el archivo se usa cuando se ejecuta la aplicación, y no se tiene que estar compilando constantemente cuando es ejecutada.

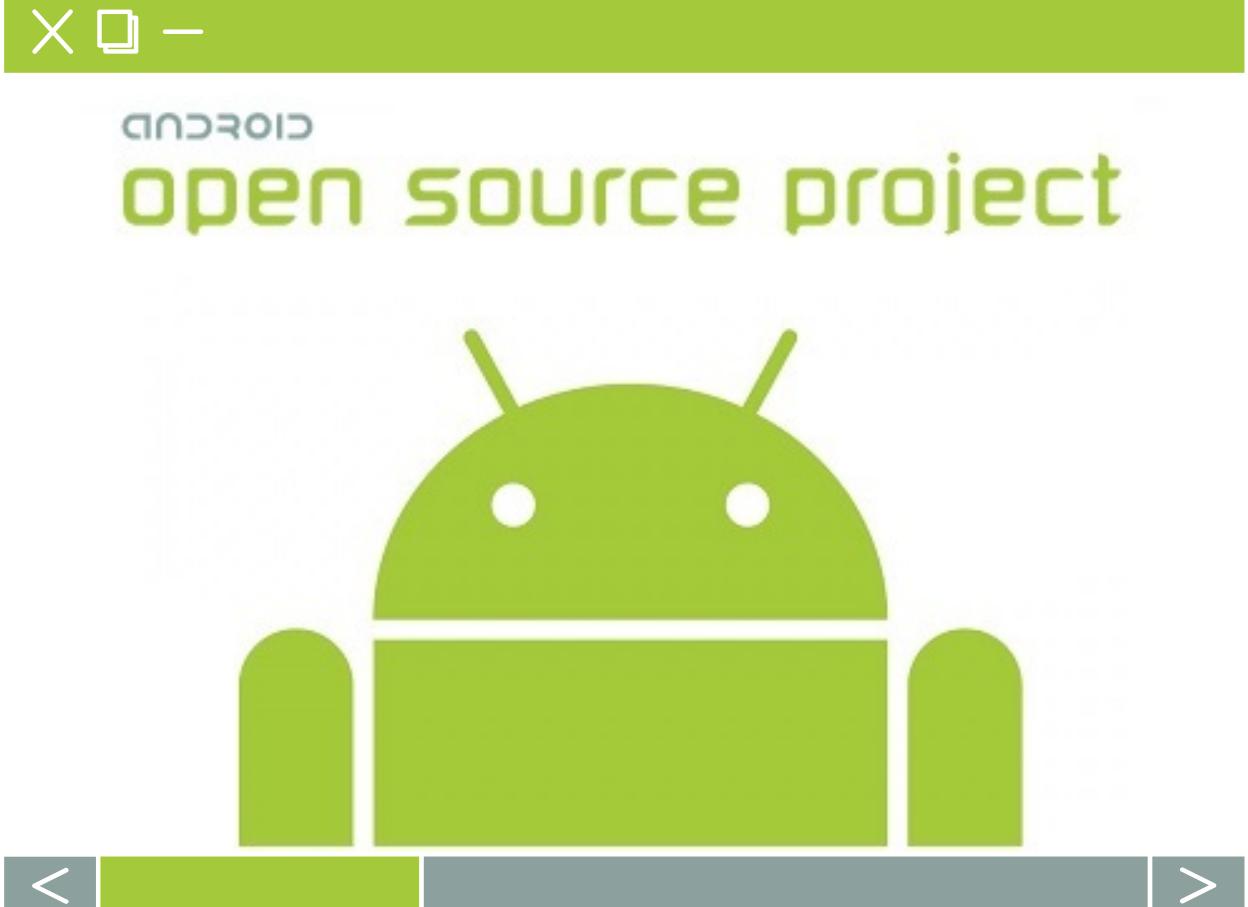
Debido a este cambio Android no lleva a cabo una cantidad tan amplia de compilaciones por cada aplicación. El resultado es que el uso de la CPU se reduce aumentando el rendimiento del hardware y el ahorro de batería.

¿QUE ES AOSP?

Android Open Source Project, es un proyecto de sistema operativo de código abierto mantenido por Google.

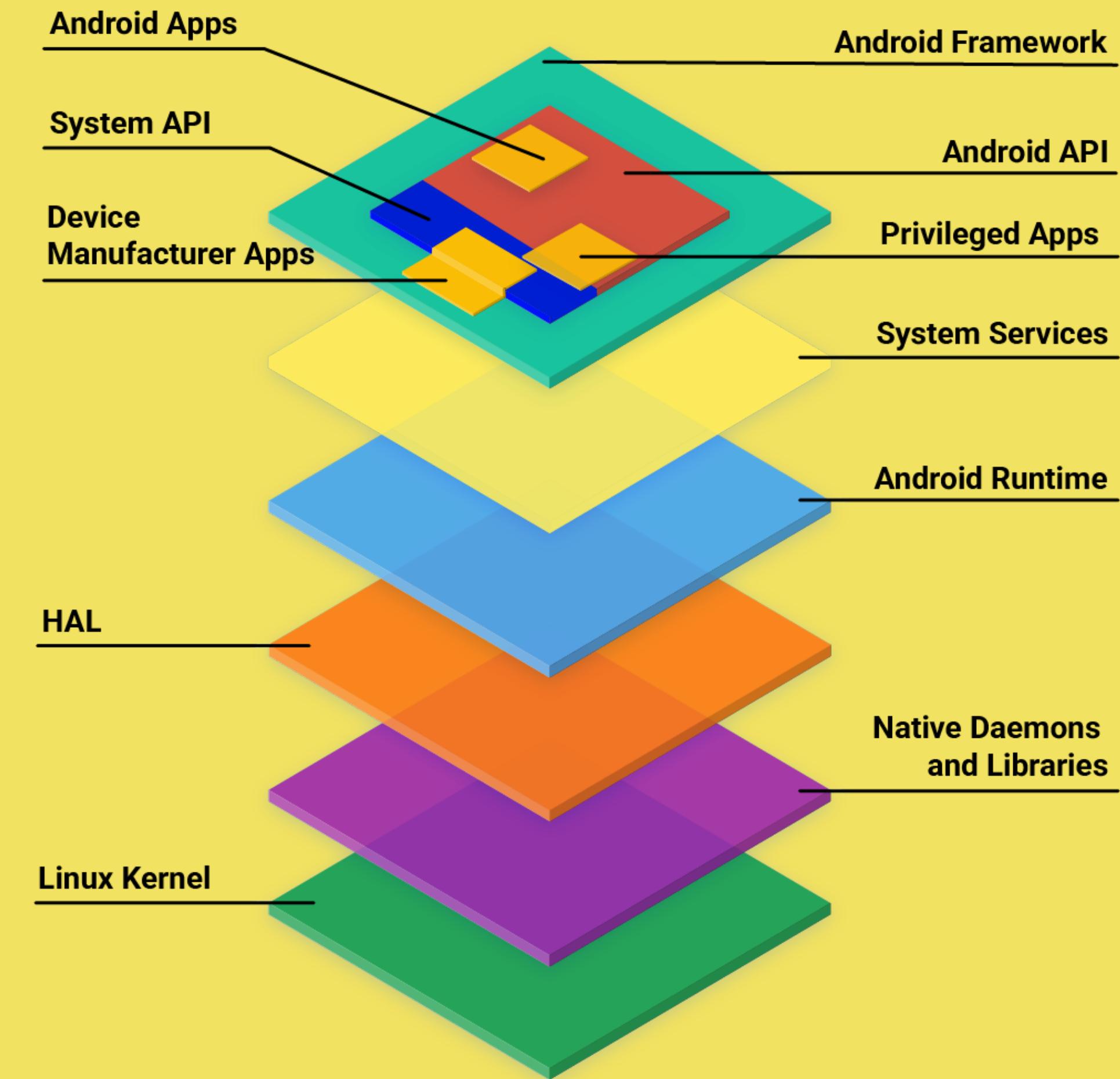
AOSP es un compendio de herramientas que permiten a los desarrolladores crear una versión de Android (código base, documentación y pruebas), pero por sí solo no es suficiente para poner en funcionamiento un móvil.

Cualquier persona puede acceder y modificar el código de AOSP, contribuyendo a el proyecto. Sin embargo, Google es quién se encarga de la dirección.



ARQUITECTURA ASOP

La pila de software para AOSP contiene las siguientes capas:





Marco de trabajo Android

Conjunto de clases Java, interfaces y código precompilado sobre el que se construyen las aplicaciones. Se puede acceder públicamente a partes del marco mediante el uso de la API de Android. Otras partes del marco están disponibles solo para los OEM mediante el uso de las API del sistema. Dentro de este engloba: Aplicación Privilegiada, Aplicación del Fabricante del dispositivo, API del sistema y el API de Android.



Servicios del sistema

Los servicios del sistema son componentes modulares como: `system_server`, `SurfaceFlinger` y `MediaService`. La funcionalidad expuesta por la API del marco de trabajo de Android se comunica con los servicios del sistema para acceder al hardware subyacente.



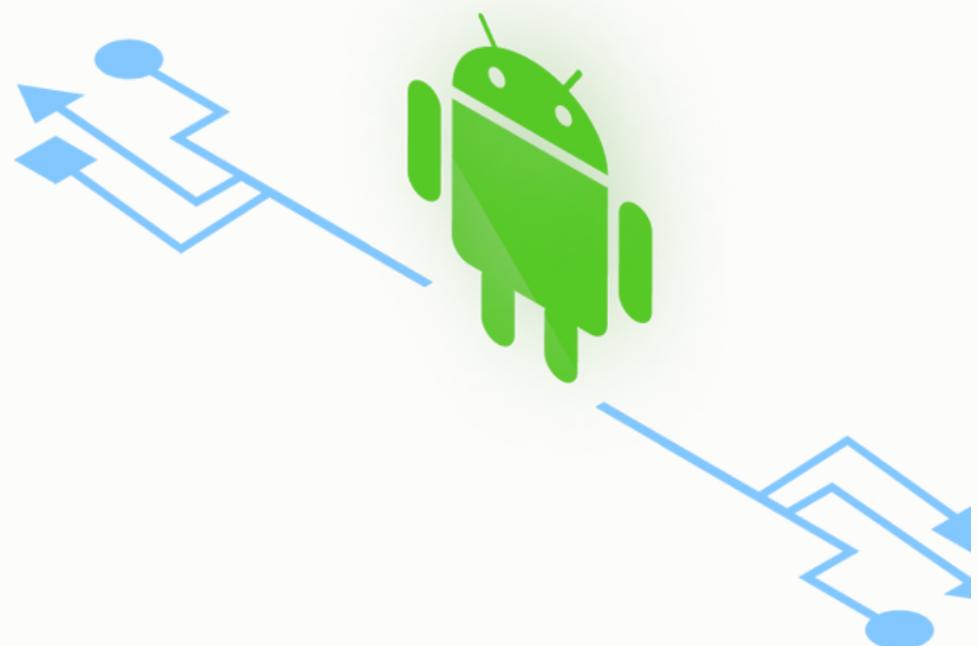
Tiempo de ejecución de Android (ART)

Un entorno de tiempo de ejecución de Java. ART realiza la traducción del código de bytes de la aplicación en instrucciones específicas del procesador que ejecuta el entorno de tiempo de ejecución del dispositivo.



Capa de abstracción de hardware (HAL)

Una HAL es una capa de abstracción con una interfaz estándar para que la implementen los proveedores de hardware. Las HAL permiten que Android sea independiente de las implementaciones de controladores de nivel inferior. El uso de una HAL le permite implementar la funcionalidad sin afectar o modificar el sistema de nivel superior.



Demonios nativos y bibliotecas

Los demonios nativos incluyen `init`, `logd` y `storaged`. Estos demonios interactúan directamente con el kernel u otras interfaces y no dependen de una implementación HAL.

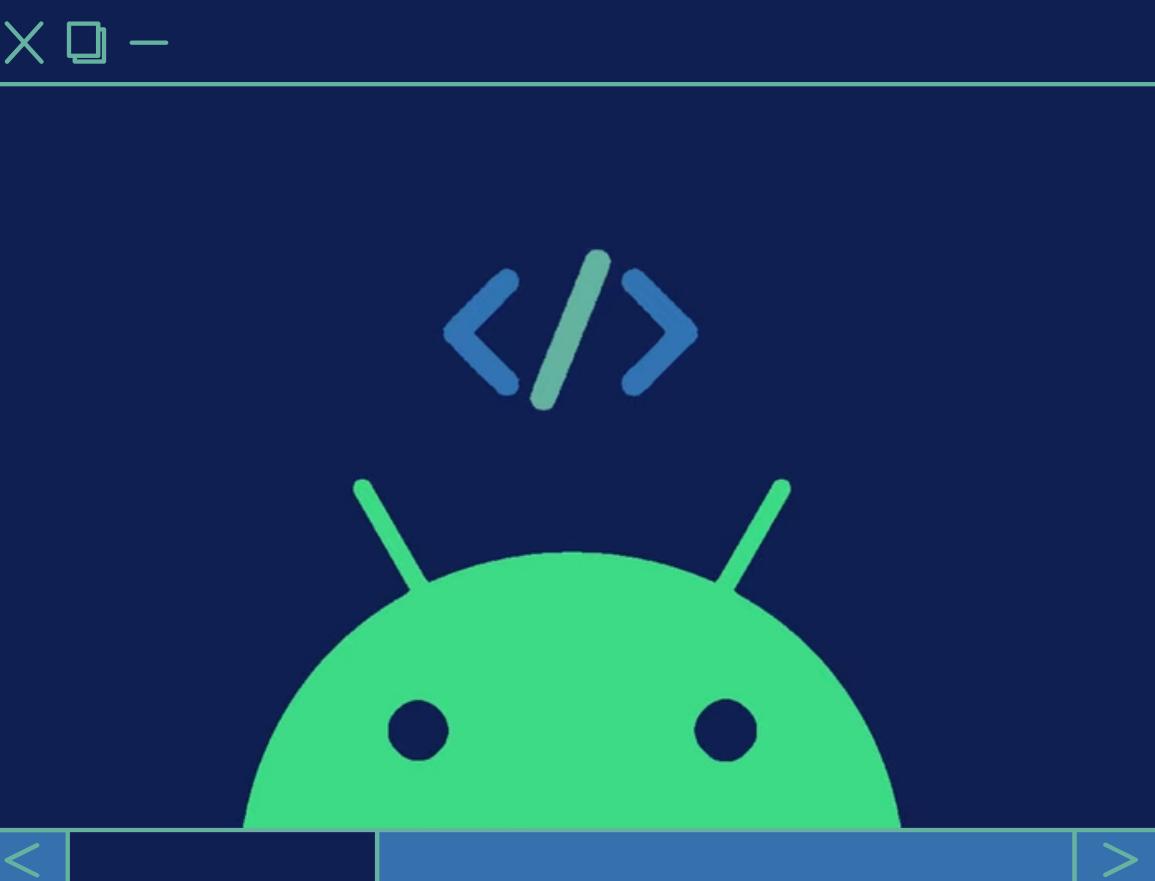
Las bibliotecas nativas incluyen `libc`, `liblog`, `libutils`, `libbinder` y `libselinux`. Estas bibliotecas nativas interactúan directamente con el kernel u otras interfaces y no dependen de una implementación de HAL.



Núcleo

El kernel es la parte central de cualquier sistema operativo, este se comunica con el hardware subyacente en un dispositivo. El núcleo AOSP se divide en módulos independientes del hardware y módulos específicos del proveedor.

¿POR QUÉ ANDROID Y AOSP NO SON LO MISMO?



El código de AOSP no cuenta con los controladores (drivers) que el núcleo de Linux necesita para manejar el hardware. Este es agregado por los fabricantes de teléfonos, chips y demás.

Los fabricantes, y los desarrolladores de ROM's, agregan sus propias características al sistema operativo, creando sus propias versiones de Android.

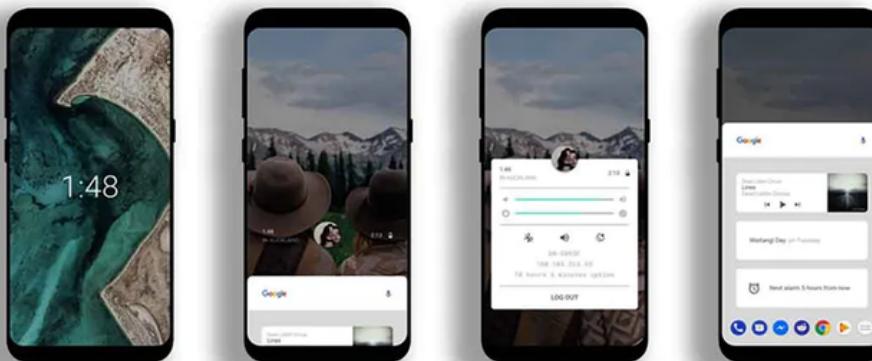
Android, por su parte, es el sistema operativo completo, y totalmente funcional en un móvil. Se crea partiendo de las bases y herramientas de AOSP, y agregando otras cosas. Pero nunca AOSP por sí solo será Android.

DERIVADOS DE ANDROID



cyanogenmod

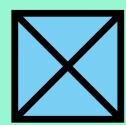
LINEAGE OS



CyanogenMod: como CM, es un sistema operativo descontinuado de código abierto desarrollado por Cyanogen para dispositivos móviles, basado en el sistema operativo Android e incluyendo código adicional que fue desarrollado como software libre y de código abierto. Sus servicios dejaron de estar operativos a partir del 31 de diciembre de 2016, siendo descontinuado y sustituido en su lugar por el proyecto LineageOS.

LineageOS: una herencia directa de CyanogenMod. Al entrar a la web oficial encontramos la siguiente frase La personalización es primordial para la productividad: Es por eso que LineageOS promete impulsar la personalización y preferencia del usuario. Todo el mundo es único y su dispositivo debe serlo también.

REFERENCIAS



- Bhardwaj, R. (2021, 26 agosto). Mobile Operating System vs Desktop Operating System. IP With Ease. <https://ipwitthease.com/mobile-operating-system-vs-desktop-operating-system/>
- De la Calle, R. (2022, 13 julio). ¿Qué sistemas operativos móviles han existido más allá de Android y de iOS? MovilZona. <https://www.movilzona.es/2021/06/30/sistemas-operativos-moviles/>
- Sacristán, L. (2021, 30 diciembre). BlackBerry OS llega a su fin en 2022: la compañía abandonará el soporte de su sistema operativo en unos . . . Xataka Móvil. <https://www.xatakamovil.com/blackberry/blackberry-os-llega-a-su-fin-2022-compania-abandonara-soporte-su-sistema-operativo-unos-dias>
- Redacción Conceptodefinicion.de. (2019, 23 julio). Symbian. Concepto de - Definición de. <https://conceptodefinicion.de/symbian/>
- Revelo, J. (2020, 30 noviembre). Aprendiendo Sobre La Arquitectura De Android. Developou. <https://www.developou.com/aprendiendo-la-arquitectura-de-android/>



- GeeksforGeeks. (2022, 2 diciembre). Difference between Windows and Android. <https://www.geeksforgeeks.org/difference-between-windows-and-android/>
- Gonzalez, G. (s.f.) ¿Qué significa que Android esté basado en Linux?. ThinkBig <https://blogthinkbig.com/android-esta-basado-en-linux>
- Reyes, P.(2020, 2 marzo) Despeja todas tus dudas sobre AOSP y descubre por qué no es lo mismo que Android. <https://androidphoria.com/desarrollo/aosp-que-es>
- Source Android (s.f.) Descripción general de la arquitectura. <https://source.android.com/docs/core/architecture?hl=es-419>
- Gonzales, C. (s.f.) ¿Qué es ART? Android Runtime, el sucesor de Dalvik. <https://androidayuda.com/android/que-es/art-android-runtime/>