

Universidad de Costa Rica

Escuela de Ciencias de la Computación e Informática

Sistemas Operativos

Android y Linux: Algunas diferencias

Juan José Villalobos Soto B06882

Grupo 3

Noviembre 2013

Introducción

Android es un sistema operativo para dispositivos móviles que ha gozado de gran popularidad, sobre todo en los últimos años, ganándole terreno incluso al iOS de Apple en los últimos años como primera opción de sistema operativo para móviles. Está basado sobre un kernel de Linux, sin embargo, al ser un sistema para móviles, presenta algunas diferencias significativas respecto al kernel Linux que se usa en PC. A continuación se comentarán las principales diferencias que presenta Android contra Linux en cuanto a manejo de procesos se refiere.

Historia de Android

Android surgió como una idea para desarrollar un sistema operativo open source para móviles a inicios de la década de los 2000, en una empresa llamada Android Inc., fundada por Andy Rubin, quien anteriormente había trabajado para Microsoft y Apple. Google en 2005 vio potencial en la empresa y la compró, colocando a Rubin como vicepresidente de ingeniería en Google y a cargo del desarrollo del sistema operativo, que pasó a llamarse Android en honor a la empresa que gestó sus inicios.

Este sistema operativo se dio a conocer en 2007 cuando fue anunciado al público, y despertando interés y negociaciones con distintos fabricantes de teléfonos para que usaran el sistema operativo, con la idea de competir contra el iOS de Apple y el Symbian de Nokia, entre otros. Fue la empresa HTC la que lanzó al mercado el primer dispositivo Android, el HTC Dream en diciembre de 2008. Desde ese momento, Android ha logrado apoderarse de una gran parte del mercado, al punto que hoy en día es el sistema operativo para móviles más usado en muchas partes del mundo, y cada vez más fabricantes optan por usarlo. [7]

Estructura de Android

Android está basado en el kernel de Linux 2.6, sin embargo, por tratarse de dispositivos móviles, presenta algunas diferencias significativas respecto a Linux puro, esto se debe principalmente a que los dispositivos móviles tienen limitaciones en cuanto a la alimentación de poder se refiere, es por ello que a la hora del manejo de procesos y memoria, se hace de manera un poco distinta a Linux.

[1][6].

En la siguiente figura se muestra la estructura que posee Android como sistema operativo



Como se puede apreciar, en la primera capa se tiene el kernel de Linux modificado, donde cabe destacar la presencia del Binder, que es el medio que utilizan los procesos para comunicarse entre ellos en este sistema[5].

Binder y manejo de procesos

El Binder es el mecanismo de comunicación entre procesos que posee Android. Es utilizado cuando los procesos se crean, o cuando una (o varias) rutinas de un proceso hacen llamado a rutinas de otro proceso, el Binder se utiliza como medio de intercambio de mensajes y parámetros entre procesos. Es importante señalar que Android sólo utiliza el Binder para la comunicación interprocesal, no posee los mecanismos nativos de Linux en este aspecto.[5]

Ejemplos de uso del Binder vale la pena mencionar que es el medio en el cual el manager de ventanas interactúa con los clientes de dicha ventana. También el activity manager utiliza el binder para lanzar procesos nuevos y componentes de procesos. Incluso el Binder está implicado en asuntos de seguridad en Android, debido a que el manejo de mensajes y parámetros lo hace con el uso de tokens, que siempre pueden ser verificadas para mantener la integridad del sistema. [5]

Planificación de Procesos

Android, al estar basado en Linux, utiliza un algoritmo de planificación conocido como CFS, o Completely Fair Scheduler (Planificador completamente justo , en inglés). Este algoritmo diferencia dos grupos principales del procesos: ForeGround y BackGround.[2][1]

Los procesos en Foreground utilizan un 80% del tiempo de CPU, mientras que los Background utilizan el 20% restante. Los procesos que están en la cola de Foreground tienen asignación de tiempo de ejecución en Round Robin con prioridad(igual cantidad de tiempo para procesos de la misma prioridad), mientras que los procesos de fondo (background) se les da tiempo de ejecución usando una cola FIFO(First In, First Out). [2]

A partir de Android 2.2, los procesos que no se estén usando actualmente, son pasados a la cola de background, donde no consumen recursos, pero siguen cargados en memoria, esto con el fin de ahorrar tiempos de lanzamiento(en caso de que la aplicación se utilice de nuevo), así como de ahorrar

energía, debido a que lanzar un proceso nuevo desde cero consume más energía que “despertar” un proceso inactivo, lo cual extiende la vida de la batería.[6].

Otras diferencias entre Android y Linux

Otras diferencias significativas entre el Kernel de Linux y Android se pueden ver en las bibliotecas de manejo de memoria: Android utiliza su propia versión de SHMEM (biblioteca de memoria compartida de POSIX), denominada ASHMEM (Android Shared Memory), la cual está basada fuertemente en SHMEM, sin embargo utiliza una API basada en archivos más sencilla de utilizar.[4]

También utiliza una biblioteca propia para alojamiento de memoria llamada Pmem (precoess memory), la cual en principio, es muy similar a ASHMEM, sin embargo, pmem se utiliza para reservar regiones de memoria contiguas relativamente grandes, y se utiliza en vez de ASHMEM cuando se necesitan búfers de memoria contiguos y no compartidos. Además pmem mantiene un mapeo de memoria virtual a física para funcionar adecuadamente. [4]

La diferencia principal que presenta Android sobre Linux es el uso de una máquina virtual para correr los procesos. Esta máquina virtual se llama Dalvik. Android utiliza aplicaciones escritas en Java, y Dalvik traduce el bytecode de java a su propio formato nativo (el .dex) antes de instalar un programa en un dispositivo. La utilización de la máquina virtual Dalvik está optimizado para dispositivos con limitaciones de hardware en aspectos de memoria y velocidad de procesador(lo que la hace ideal para dispositivos móviles).[8]

La diferencia principal entre la máquina de Java y la Dalvik es que la máquina de Java es basada en pila, mientras que la Dalvik está basada en registros. La máquina Dalvik posee una herramienta llamada dx que traduce de Java bytecode a .dex, que es un formato más compacto que el

.class de Java, esto porque strings duplicados y constantes sólo se guardan una vez en el .dex, lo que reduce el tamaño del archivo en un pequeño porcentaje. Así como el uso de operaciones de 16 bits, en contraste a Java que usa 8 bits, y también se permite correr varias instancias de la máquina virtual al mismo tiempo. [8]

Conclusiones

Pese a que Android, está basado en Linux, hay diferencias significativas entre ambos sistemas. Esto se da principalmente porque Android es un sistema para dispositivos móviles, los cuales tienen limitaciones de memoria y procesador. Por esto, se han diseñado mecanismos distintos para el manejo de procesos y memoria, con el fin de aprovechar al máximo los limitados recursos del dispositivo.

Es importante mencionar que dichas diferencias dan paso a una serie de aplicaciones nuevas a estos mecanismos, por ejemplo, se teoriza utilizar los mecanismos de planificación de procesos de Android para implementar un scheduling en el sistema de mensajes de texto similar, con prioridades de contactos y demás. [1]

Estas ideas nuevas permiten desarrollar nuevas tecnologías que pueden ser aprovechadas por todos nosotros en un futuro.

Referencias

1) Priority Based Pre-emptive Task Scheduling for Android Operating System

<http://arxiv.org/ftp/arxiv/papers/1304/1304.7889.pdf>

2) Linux Scheduling <http://www.cs.columbia.edu/~krj/os/lectures/L12-LinuxSched.pdf>

3) How Android works the Big Picture

<http://www.zdnet.com/blog/burnette/how-android-works-the-big-picture/515>

4) Android Kernel Features http://elinux.org/Android_Kernel_Features

5) Android Binder http://elinux.org/Android_Binder

6) Android Task Killers

<http://www.phonedog.com/2011/06/26/the-truth-about-android-task-killers-and-why-you-don-t-need-them/>

7) Historia y comienzos de Android

<http://www.elandroidlibre.com/2011/08/la-historia-y-los-comienzos-de-android-el-sistema-operativo-de-google.html>

8) Dalvik Virtual Machine http://en.wikipedia.org/wiki/Dalvik_virtual_machine