

URI	<i>Uniform Resource Identifier</i>
USB	<i>Universal Serial Bus</i>
W3C	<i>World Wide Web Consortium</i>
WSDL	<i>Web Services Description Language</i>
WWW	<i>World Wide Web</i>
XML	<i>Extensible Markup Language</i>

CAPÍTULO 1.

Visión general y entorno de desarrollo

La telefonía móvil está cambiando la sociedad actual de una forma tan significativa como lo ha hecho Internet. Esta revolución no ha hecho más que empezar, los nuevos terminales ofrecen unas capacidades similares a un ordenador personal, lo que permite que puedan ser utilizados para leer nuestro correo o navegar por Internet. Pero a diferencia de un ordenador, un teléfono móvil siempre está en el bolsillo del usuario. Esto permite un nuevo abanico de aplicaciones mucho más cercanas al usuario. De hecho, muchos autores coinciden en que el nuevo ordenador personal del siglo veintiuno será un terminal móvil.

El lanzamiento de Android como nueva plataforma para el desarrollo de aplicaciones móviles ha causado una gran expectación y está teniendo una importante aceptación tanto por los usuarios como por la industria. En la actualidad se está convirtiendo en una seria alternativa a otras plataformas como Symbian, iPhone o Windows Mobile.

A lo largo de este capítulo veremos las características de Android, que lo hacen diferente de sus competidores. Se explicará también cómo instalar y trabajar con el entorno de desarrollo (Eclipse + Android SDK).

1.1. ¿Qué hace Android especial?

Como hemos comentado, existen muchas plataformas para móviles (iPhone, Symbian, Windows Phone, BlackBerry, Palm, Java Mobile Edition, Linux Mobile (LiMo),...); sin embargo Android presenta una serie de características que lo hacen diferente. Es el primero que combina en una misma solución las siguientes cualidades:

- Plataforma realmente abierta. Es una plataforma de desarrollo libre basada en Linux y de código abierto. Una de sus grandes ventajas es que se puede usar y customizar el sistema sin pagar *royalties*.
- Portabilidad asegurada. Las aplicaciones finales son desarrolladas en Java lo que nos asegura que podrán ser ejecutadas en gran variedad de dispositivos, tanto presentes como futuros. Esto se consigue gracias al concepto de máquina virtual.
- Arquitectura basada en componentes inspirados en Internet. Por ejemplo, el diseño de la interfaz de usuario se hace en xml, lo que permite que una misma aplicación se ejecute en un móvil de pantalla reducida o en un netbook.
- Filosofía de dispositivo siempre conectado a Internet.
- Gran cantidad de servicios incorporados: por ejemplo, localización basada tanto en GPS como en torres de telefonía móvil. Incorpora potentes bases de datos con SQL. Reconocimiento y síntesis de voz, navegador, mapas...
- Alto nivel de seguridad. Los programas se encuentran aislados unos de otros gracias al concepto de ejecución dentro de una caja que incorpora la máquina virtual. Cada aplicación dispone de una serie de permisos que limitan su rango de actuación (servicios de localización, acceso a Internet...)
- Optimización para baja potencia y poca memoria. Por ejemplo, Android utiliza la Máquina Virtual Dalvik. Se trata de una implementación de Google de la máquina virtual de Java optimizada para dispositivos móviles.
- Alta calidad de gráficos y sonido: gráficos vectoriales suavizados, animaciones inspiradas en Flash, gráficos en 3 dimensiones basados en OpenGL. Incorpora codecs estándar más comunes de audio y vídeo, incluyendo H.264 (AVC), MP3, AAC,...

Como hemos visto Android combina características muy interesantes. No obstante, la pregunta del millón es ¿Se convertirá Android en el nuevo estándar de sistema operativo (S.O.) para móviles? Para contestar a esta pregunta habrá que esperar un tiempo y ver cuál es la respuesta de Windows con el lanzamiento de su nuevo S.O. para móviles.

En conclusión Android nos ofrece una forma sencilla y novedosa de implementar potentes aplicaciones para móviles. A lo largo de este texto trataremos de mostrar de la forma más sencilla posible como conseguirlo.

1.2. Los orígenes

Google adquiere Android Inc. en el año 2005. Se trataba de una pequeña compañía, que acababa de ser creada, orientada a la producción de aplicaciones para terminales móviles. Ese mismo año empiezan a trabajar en la creación de una máquina virtual Java optimizada para móviles (Dalvik VM).

En el año 2007 se crea el consorcio Handset Alliance¹ con el objetivo de desarrollar estándares abiertos para móviles. Está formado por Google, Intel, Texas Instruments, Motorola, T-Mobile, Samsung, Ericson, Toshiba, Vodafone, NTT DoCoMo, Sprint Nextel y otros. Una pieza clave de los objetivos de esta alianza es promover el diseño y difusión de la plataforma Android. Sus miembros se han comprometido a publicar una parte importante de su propiedad intelectual como código abierto bajo licencia Apache v2.0.

En noviembre del 2007 se lanza una primera versión del Android SDK. Al año siguiente aparece el primer móvil con Android (T-Mobile G1). En octubre Google libera el código fuente de Android principalmente bajo licencia de código abierto Apache (licencia GPL v2 para el núcleo). Ese mismo mes se abre Android Market, para la descarga de aplicaciones. En abril del 2009 Google lanza la versión 1.5 del SDK que incorpora nuevas características como el teclado en pantalla. A finales del 2009 se lanza la versión 2.0 y durante el 2010 las versiones 2.1, 2.2 y 2.3.

Durante el año 2010 Android se consolida como uno de los sistemas operativos para móviles más utilizados, con resultados cercanos al iPhone e incluso superando al sistema de Apple en EE.UU.

1.3. Comparativa con otras plataformas

En este apartado vamos a describir las características de las principales plataformas móviles disponibles en la actualidad. Dado la gran cantidad de datos que se indican, hemos utilizado una tabla para representar la información. De esta forma resulta más sencillo comparar las plataformas.

¹ <http://www.openhandsetalliance.com>



	Apple iOS 4.2	Android 2.3	Symbian^3 OS	Windows Phone 7	BlackBerry OS 6
Compañía	Apple	Open Handset Alliance	Symbian Foundation	Windows	RIM
Núcleo del SO	Mac OS X	Linux	Mobile OS	Windows CE	Mobile OS
Familia CPU soportada	ARM	ARM, MIPS, Power, x86	ARM	ARM	ARM
Lenguaje de programación	Objective-C, C++	Java, C++	C++	C++	Java
Licencia de software	propietaria	software libre	software libre	propietaria	propietaria
Edad de la plataforma	adolescente	joven	Madura	joven	madura
Motor del navegador web	WebKit	WebKit	WebKit	Pocket Internet Explorer	WebKit
Soporte Flash	No	Sí	Sí	No	No
HTML5	Sí	Sí	No	No	Sí
Tienda de aplicaciones	App Store	Android Market	Ovi Store	Windows Marketplace	BlackBerry App World
Número de aplicaciones	300.000 30% gratuitas	140.000 60% gratuitas	13.000	4.000	15.000
Facilidad uso	excelente	excelente	buena	excelente	buena
Interfaz personalizable	No	Sí	Sí	Sí	Sí
Actualizaciones automáticas del S.O.	Sí	depende del fabricante	Sí	¿?	Sí
Soporte memoria externa	No	Sí	Sí	No	Sí
Fabricante único	Sí	No	No	No	Sí
Variedad de dispositivos	modelo único	muy alta	muy alta	baja	baja
Tipo de pantalla	capacitativa	capacitiva /resistiva	capacitiva /resistiva	capacitativa	capacitiva /resistiva
Aplicaciones nativas	Sí	Sí	Sí	Sí	No

Tabla 1: Comparativa de las principales plataformas móviles

Otro aspecto fundamental a la hora de comparar las plataformas móviles es su cuota de mercado. En la siguiente gráfica podemos ver un estudio realizado por la empresa Gratner Group, donde se muestra la evolución del mercado de los sistemas operativos para móviles según el número de terminales vendidos. Podemos destacar el importante descenso de ventas de Nokia, además del ascenso continuado de plataforma Android.

Se han realizado otros tipos de estudios que miden la actividad de los usuarios en Internet. En estos casos se comprueba como los usuarios de Android y iPhone son los más activos, mientras que los usuarios con otras plataformas, como Symbian, utilizan sus terminales de forma más convencional.

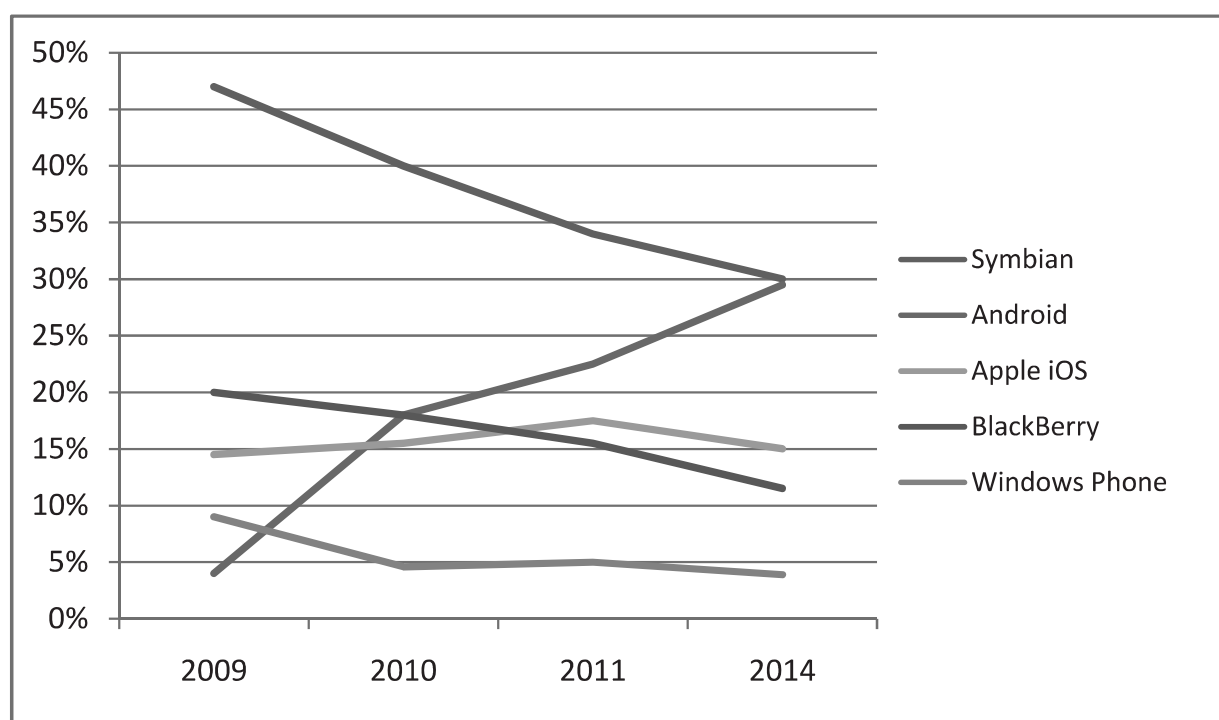


Figura 1: Cuota de mercado de los sistemas operativos para móviles hasta el 2011 y previsión para el 2014 en el mundo (fuente: Gratner Group).

1.4. Arquitectura de Android

El siguiente gráfico muestra la arquitectura de Android. Como se puede ver está formada por cuatro capas. Una de las características más importantes es que todas las capas están basadas en *software* libre.

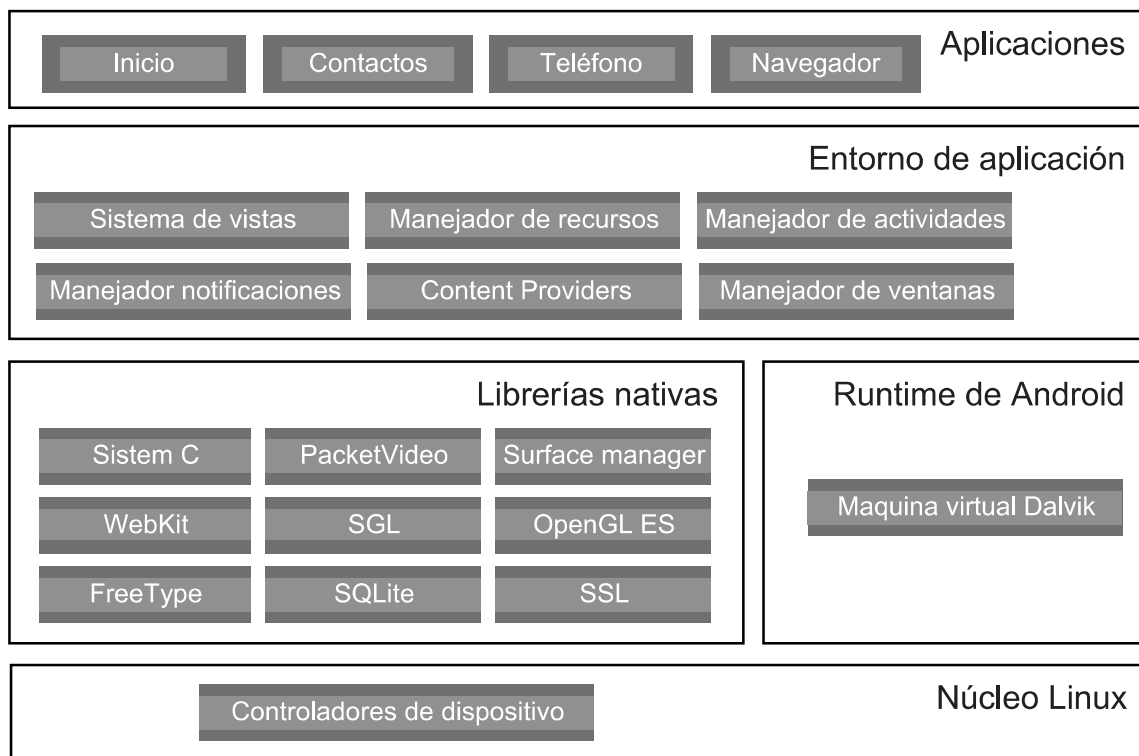


Figura 2: Arquitectura de Android.

1.4.1. El núcleo Linux

El núcleo de Android está formado por el sistema operativo Linux versión 2.6. Esta capa proporciona servicios como la seguridad, el manejo de la memoria, el multiproceso, la pila de protocolos y el soporte de *drivers* para dispositivos.

Esta capa del modelo actúa como capa de abstracción entre el hardware y el resto de la pila. Por lo tanto, es la única que es dependiente del *hardware*.

1.4.2. *Runtime* de Android

Está basado en el concepto de máquina virtual utilizado en Java. Dado las limitaciones de los dispositivos donde ha de correr Android (poca memoria y procesador limitado) no fue posible utilizar una máquina virtual Java estándar. Google tomó la decisión de crear una nueva, la máquina virtual Dalvik, que respondiera mejor a estas limitaciones.

Algunas características de la máquina virtual Dalvik que facilitan esta optimización de recursos son: que ejecuta ficheros Dalvik ejecutables (.dex) – formato optimizado para ahorrar memoria. Además, está basada en registros. Cada aplicación corre en su propio proceso Linux con su propia instancia de la máquina virtual Dalvik. Delega al kernel de Linux algunas funciones como *threading* y el manejo de la memoria a bajo nivel.

También se incluye en el *Runtime de Android* el “core libraries” con la mayoría de las librerías disponibles en el lenguaje Java.

1.4.3. Librerías nativas

Incluye un conjunto de librerías en C/C++ usadas en varios componentes de Android. Están compiladas en código nativo del procesador. Muchas de las librerías utilizan proyectos de código abierto. Algunas de estas librerías son:

- **System C library:** una derivación de la librería BSD de C estándar (libc), adaptada para dispositivos embebidos basados en Linux.
- **Media Framework:** librería basada en PacketVideo's OpenCORE; soporta codecs de reproducción y grabación de multitud de formatos de audio vídeo e imágenes MPEG4, H.264, MP3, AAC, AMR, JPG y PNG.
- **Surface Manager:** maneja el acceso al subsistema de representación gráfica en 2D y 3D.
- **WebKit:** soporta un moderno navegador web utilizado en el navegador Android y en la vista webview. Se trata de la misma librería que utiliza Google Chrome y Safari de Apple.
- **SGL:** motor de gráficos 2D.
- **Librerías 3D:** implementación basada en OpenGL ES 1.0 API. Las librerías utilizan el acelerador hardware 3D si está disponible, o el software altamente optimizado de proyección 3D.
- **FreeType:** fuentes en bitmap y renderizado vectorial.
- **SQLite:** potente y ligero motor de bases de datos relacionales disponible para todas las aplicaciones.
- **SSL:** proporciona servicios de encriptación *Secure Socket Layer*.

1.4.4. Entorno de aplicación

Proporciona una plataforma de desarrollo libre para aplicaciones con gran riqueza e innovaciones (sensores, localización, servicios, barra de notificaciones,). También se conoce como Java SDK.

La arquitectura ha sido diseñada para simplificar la reutilización de componentes. Las aplicaciones pueden publicar sus capacidades y otras pueden hacer uso de ellas (sujetas a las restricciones de seguridad). Este mismo mecanismo permite a los usuarios reemplazar componentes.

Una de las mayores fortalezas del entorno de aplicación de Android es que se aprovecha el lenguaje de programación Java. El SDK de Android no acaba de ofrecer todo lo disponible para su estándar del entorno de ejecución Java (JRE), pero es compatible con una fracción muy significativa de la misma.

Los servicios más importantes que incluye son:

- **Views:** extenso conjunto de vistas, (parte visual de los componentes).
- **Resource Manager:** proporciona acceso a recursos que no son en código.
- **Activity Manager:** maneja el ciclo de vida de las aplicaciones y proporciona un sistema de navegación entre ellas.
- **Notification Manager:** permite a las aplicaciones mostrar alertas personalizadas en la barra de estado.
- **Content Providers:** mecanismo sencillo para acceder a datos de otras aplicaciones (como los contactos).

1.4.5. Aplicaciones

Este nivel está formado por el conjunto de aplicaciones instaladas en una máquina Android. Todas las aplicaciones han de correr en la máquina virtual Dalvik para garantizar la seguridad del sistema.

Normalmente las aplicaciones Android están escritas en Java. Para desarrollar aplicaciones en Java podemos utilizar el Android SDK. Existe otra opción consistente en desarrollar las aplicaciones utilizando C/C++. Para esta opción podemos utilizar el Android NDK (Native Development Kit).

1.5. Instalación del entorno de desarrollo

Para el desarrollo de las aplicaciones vamos a poder utilizar un potente y moderno entorno de desarrollo. Al igual que Android, todas las herramientas están basadas en *software* libre. Existen varias alternativas para usar como entorno de desarrollo. En este texto se supondrá que estamos trabajando con el *software* enumerado a continuación:

- Java 5.0 o superior.
- Eclipse (Eclipse IDE for Java Developers).
- Android SDK (Google).
- Eclipse Plug-in (Android Development Toolkit- ADT).

Describiremos a continuación el proceso a seguir para instalar el *software* anterior.

1.5.1. Instalación de la máquina virtual Java

Este *software* va a permitir ejecutar código Java en tu equipo. A la máquina virtual Java también se la conoce como entorno de ejecución Java, Java Runtime Environment (JRE) o Java Virtual Machine (JVM).

Muy posiblemente ya tengas instalada la Máquina Virtual Java en tu equipo. Si es así puedes pasar directamente al punto siguiente. En caso de dudas, puedes pasar también al punto siguiente. Al concluirlo te indicará si la versión de la máquina virtual Java es incorrecta. En caso necesario, regresa a este punto para instalar una adecuada.

Para instalar la Máquina Virtual Java accede a <http://java.com/es/download/> y descarga e instala el fichero correspondiente. En Windows el archivo es `jxpiinstall.exe`.

1.5.2. Instalación de Eclipse

El primer paso va a consistir en instalar Sun JDK y Eclipse. La versión del JDK ha de ser 5.0 o superior. Nosotros usaremos Sun JDK 6.0. Eclipse resulta el entorno de desarrollo más recomendable para Android, es libre y además es soportado por Google (ha sido utilizado por los desarrolladores de Google para crear Android). La versión mínima es la 3.3.1.

Para instalar Eclipse y Sun JDK hay que seguir los siguientes pasos:

- 1) Accede a la página <http://www.eclipse.org/downloads/> y descarga la última versión de “Eclipse IDE for Java Developers”. Verás que se encuentra disponible para los sistemas operativos más utilizados, como Windows, Linux y Mac OS.

NOTA: en este texto hemos utilizado la versión Helios para Windows 32 bits (fichero: “*eclipse-java-helios-win32.zip*”).

- 2) Este *software* no requiere una instalación específica, simplemente descomprimir los ficheros en la carpeta que prefieras. Si así lo deseas puedes crear un acceso directo en el escritorio o en el menú inicio.