CHAPTER 1

History and Evolution of the Android OS

> *I'm going to destroy Android, because it's a stolen product. I'm willing to go thermonuclear war on this.*
>
> —Steve Jobs, Apple Inc.

Android, Inc. started with a clear mission by its creators. According to Andy Rubin, one of Android's founders, Android Inc. was to develop "smarter mobile devices that are more aware of its owner's location and preferences." Rubin further stated, "If people are smart, that information starts getting aggregated into consumer products." The year was 2003 and the location was Palo Alto, California. This was the year Android was born.

While Android, Inc. started operations secretly, today the entire world knows about Android. It is no secret that Android is an operating system (OS) for modern day smartphones, tablets, and soon-to-be laptops, but what exactly does that mean? What did Android used to look like? How has it gotten where it is today? All of these questions and more will be answered in this brief chapter.

Origins

Android first appeared on the technology radar in 2005 when Google, the multibillion-dollar technology company, purchased Android, Inc. At the time, not much was known about Android and what Google intended on doing with it. Information was sparse until 2007, when Google announced the world's first truly open platform for mobile devices.

The First Distribution of Android

On November 5, 2007, a press release from the Open Handset Alliance set the stage for the future of the Android platform. The alliance stated some of the goals of Android as, "fostering innovation on mobile devices and giving consumers a far better user experience than much of what is available on today's mobile platforms."

At that time, more than 2 billion mobile phones were used worldwide, compared to the 4.6 billion used as of 2010. However, there was no coordination of platforms between the various companies that provided mobile devices. With the introduction of Android, a single operating system removed the need for reimplementation of phone applications and middleware. The companies creating new devices could now focus much more intently on the hardware and underlying components.

But these companies weren't the only ones who benefited from the launch of Android; software developers could now release applications to multiple devices with very few changes to the underlying code base. This allowed developers to spend more time working on the applications these phones were running and create the rich and impressive applications that we are all used to. This was in part due to the open source philosophy behind Android, and the Apache license, which is the license used on most of the Android source code.

Open Source Apache License

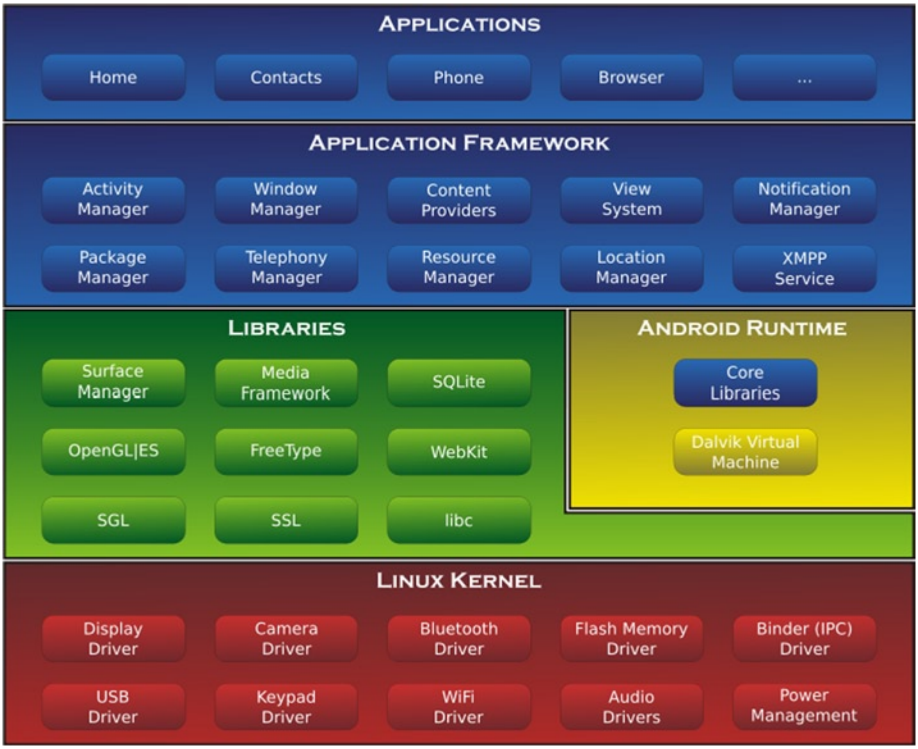
The Apache License is just one of many different licenses that exist in the open source community. While there are differences in all of these licenses, they all facilitate the same open source mindset that is best summed up as follows:

> *"Free software" is a matter of liberty, not price. To understand the concept, you should think of "free" as in "free speech," not as in "free beer."*
>
> —Richard M. Stallman

The Apache License specifically grants freedom to use the software for any purpose, as well as the ability to distribute, modify, or distribute modified versions. The Apache License is also permissive, meaning that modified versions do not have to succumb to the Apache License. For more information about the Apache License, go to `http://www.apache.org/licenses/LICENSE-2.0`.

What Is Android?

So what exactly is Android? Android OS is the open source technology stack that runs on over 400 million devices worldwide. This technology stack consists of various components that allow developers and device manufacturers to work independently. This can be broken into five primary pieces—applications, application frameworks, native libraries, Android runtime, and the Linux kernel—as shown in Figure 1-1.



Android OS (Wikipedia) *`http://en.wikipedia.org/wiki/File:Android-System-Architecture.svg`*

Figure 1-1. *The Android System Architecture*

Applications

The applications exist at the highest level. These are the tools that everyone who uses Android is most familiar with. Android comes with various robust applications that support everyday phone needs, such as messaging, e-mail, Internet browsing, and various third-party applications. These applications are primarily written in the Java programming language. In a recent legal case with Oracle, Google's Android chief Andy Rubin explained why he chose Java as the language of use for developers. Rubin's primary points were that Java had a well-known brand name and that Java is taught in almost all universities worldwide.

These applications are distributed through various means, most commonly from the Google Play Store (formerly Android Marketplace); however, the Android OS also supports installation of applications over a USB connection and from an SD card.

Application Frameworks

Android provides developers the ability and tools to create extensive, interactive, rich graphical applications to users, and is targeted to deploy these applications to the Google Play Store. Developers have access to the same APIs that are used inside of core applications, as well as access to almost all existing Java libraries. For the development process of Android applications, consult **Chapter 6: Installing the Android SDK for Intel Application Development.**

Native Libraries

The next level is where the road diverges. The native libraries and the Android runtime exist in roughly the same space. The native libraries are compiled and preinstalled C/C++ binaries that the Android system depends on. These include all of the libraries in the green section of Figure 1-1. The following sections contain descriptions of some of the more prominent native libraries and their functions inside of Android.

Surface Manager

This is often referred to as Android's Window Manager. Surface Manager is used for composing what any individual screen will look like. It also does some more subtle things that help Android run smoothly, such as off-screen buffering and transitions.

SQLite

This is a database used to persist information across sessions of an Android device. On Android, the SQLite database is stored inside of the device's internal memory so SD cards can be interchanged without losing device-specific information.

WebKit

WebKit allows for HTML to be rendered and displayed to Android very quickly and efficiently. This is the default browser engine in the Android system and is available to system and third-party applications.

OpenGL/ES

The OpenGL engine processes graphics in Android. OpenGL can render both 2D and 3D objects on Android. This also supports hardware acceleration on devices with dedicated graphic chips.

Android Runtime

Inside of the Android runtime are two primary components: the core Java libraries that Android provides, and the Dalvik virtual machine. The Dalvik virtual machine is Google's implementation of Java that is optimized to be used on mobile devices. The more specific differences within Dalvik are very technical and aren't covered in this book.

Linux Kernel

The last of the layers is the Linux kernel. Android was initially based on the Linux 2.6 kernel, with some optimizations for mobile use. Current versions of Android are based on the Linux 3.1 kernel. The Linux kernel provides access as close to the hardware as possible. As a result, drivers are written in the kernel space to operate as fast and as efficiently as possible. These include things like controlling the internal radios, turning on the stereo and camera, dealing with power and battery charging, and operating the physical keyboard or buttons on the device. The Linux kernel, like Android, is an open source project and is used widely, particularly on servers in enterprise environments.

The Open Handset Alliance

In November 2007, the Open Handset Alliance (OHA) was established by 34 founding members dedicated to development of open mobile standards, including Google, mobile device manufacturers, application developers, embedded systems developers, and commercialization companies. The goal of this alliance as described in the web site is as follows:

> *The Open Handset Alliance™, a group of 84 technology and mobile companies who have come together to accelerate innovation in mobile and offer consumers a richer, less expensive, and better mobile experience.*

As it stands today, OHA has 84 firms who are developing and working on the consortium's main and only project to date, Android. Thanks to the services and products offered by members of the OHA, devices and related services are produced at higher quality for a lower price.

Android Open Source Project

After the purchase of Android, Inc., the Android Open Source Project (AOSP) was created and has since been led by Google. The AOSP is in charge of the development and maintenance of the Android software stack. As stated by Google, the goal of the project is as follows:

> *The goal of the Android Open Source Project is to create a successful real-world product that improves the mobile experience for end users.*

Android is designed and maintained with backward capability in mind. This means that new devices can run applications developed all the way back to Android's Cupcake (1.5). Official support for the Android SDK only goes back to Cupcake (1.5), so applications written for pre-Cupcake devices aren't guaranteed to run on the latest Android devices.

Through the course of the AOSP, there have been many different versions of Android released for mobile devices. When new Android versions are released, mobile device owners are allowed to choose whether to upgrade their OS. With every iteration of Android, a new SDK is made available to developers and various new features are added to supported devices. Software developers need to stay mindful of the legacy features from previous versions when developing new applications.

Astro (1.0)

Astro is where Android started, released as a beta in November 2007, and released to the public in September of 2008 on the HTC Dream. Astro showed off various core features of the Android OS and included many of the apps that Android users now know and love. These include Android Market, a web browser, e-mail/Gmail, Google Maps, Messaging, Media Player, YouTube, and various others.

Cupcake (1.5)

Cupcake, released April 30, 2009, was the next major version of Android to hit the commercial markets. Cupcake was based on the Linux kernel 2.6.27 and included many new features to users and developers. The major changes were support for virtual keyboards, support for widgets on the home screen, animations added in various places, and auto-pairing and stereo support for Bluetooth-capable devices. On a humorous side note, from Cupcake and onward all Android versions to date have been named after desserts.

Donut (1.6)

On September 15, 2009, Google released Android version 1.6, named Donut. With Donut came an updated Linux kernel from 2.6.27 to 2.6.29, as well as some new features and supported devices. Major features included voice and text search of contacts/web/bookmarks, support for WVGA screens, and improvements to camera functionality and speed. Donut was the last version of Android in the 1.x series to be released.

Éclair (2.0/2.1)

Éclair was released October 26, 2009, which continued to be built on the Linux kernel version 2.6.29. With SDK version 2.0 came many new features and capabilities for both developers and consumers. Large changes were made to the way that Android looked and felt on capable devices, including significant speed improvements in many different applications. The premier device for Android 2.0 was Motorola's Droid on Verizon Wireless.

On December 3, 2009, Google updated Android to version 2.0.1 in efforts to fix some small bugs and update the API for developers. It wasn't until January 12, 2010 that Android was moved to version 2.1. Similar to the update in December, version 2.1 primarily included updates to the underlying API and bug fixes.

Froyo (2.2.x)

On May 20, 2010, the Android SDK version 2.2 (Froyo) was released with Linux kernel 2.6.32. Google's Nexus One was the first device on the market to show off Froyo and its new capabilities. Very significant features were added to Froyo, including Adobe Flash support, Android Cloud to Device Messaging, Wi-Fi hotspot functionality, and significant performance optimizations. It's important to note that the Android SDK recommends choosing Froyo as your base development version to hit the largest current user base of Android users.

Three subsequent updates were released for the Android 2.2 SDK: 2.2.1 on January 18, 2011, 2.2.2 on January 22, and 2.2.3 on November 21. These updates were primarily bug fixes and security updates to Android.

Gingerbread (2.3.x)

Gingerbread was released on the December 6, 2010 and was based on the Linux kernel 2.6.35. Similar to the Froyo release, Google's Nexus S was introduced to show off Gingerbread. Features of Gingerbread include support for WXGA and other extra-large screen sizes, improvements to the virtual keyboard, support for more internal sensors (namely gyroscopes and barometers), support for multiple and front-facing cameras, and the ability to read Near Field Communication (NFC) tags.

Five updates were released to Gingerbread, 2.3.3–7, from February to September of 2011. With these updates came various features, security updates, and bug fixes. One of the most significant features introduced was Open Accessory support, which lets a compatible device function as an USB peripheral to compatible software platforms.

Honeycomb (3.x)

In February of 2011, Honeycomb, the first tablet-only Android version, was released on the Motorola Xoom. Because Honeycomb was created specifically for tablet devices, Android was tweaked to allow for a more enjoyable experience with larger screen real estate. This included a redesign of the onscreen keyboard, a system bar to allow for quick access to notifications and navigation, multiple browser tabs to allow for easier use of the web, and support for multi-core processors.

Honeycomb has had six updates, two of which were major, through its current life cycle. The first update was Android SDK version 3.1 on May 10, 2011, and it namely added support for USB accessories such as keyboards, joysticks, and other human interface devices (HIDs). The second major SDK update was 3.2 on July 15, 2011. The most significant feature of 3.2 was compatibility display mode for Android applications that were not designed for tablets. The last four updates to Honeycomb have been minor improvements, bug fixes, and security updates.

Ice Cream Sandwich (4.0.x)

Ice Cream Sandwich (ICS) was released on October 19, 2011 and was based on the Linux kernel 3.0.1. Samsung's Galaxy Nexus was the device released with ICS as it hit public markets. ICS was packed with a multitude of features and improvements to the Android user interface (UI). Some features include a customizable launcher, a tabbed web browser, facial recognition to unlock the device, a built-in photo editor, hardware acceleration of the UI, and software buttons originally introduced in 3.x (Honeycomb). It is important to note that ICS merged version 3.x (Honeycomb) and 2.3.x (Gingerbread) into a single OS supporting both phones and tablets.

Four minor updates have since been released for ICS devices from November of 2011 to March of 2012. These updates focused on stability improvements, camera performance, and bug fixes.

Jelly Bean (4.1.x)

Jelly Bean was released on July 9, 2012 and is based on the Linux kernel 3.1.10. Asus' Nexus 7 tablet device was the flagship user of Jelly Bean. Jelly Bean released a number of improvements and performance upgrades to the UI and audio within Android. Version 4.2, released on November 13, 2012 and based on Linux kernel 3.4.0, added accessibility improvements. Version 4.3 was released on July 24, 2013, and added OpenGL ES 3.0 support for better game graphics, security enhancements, and upgraded digital rights management APIs. Other features of the Jelly Bean versions include customizable keyboard layouts, expandable notifications, application-specific notification filtering, and multichannel audio.

KitKat (4.4.x)

The last version of Android as of this writing, KitKat, was released on September 3, 2013. Its features included performance optimizations for devices with less RAM, expanded accessibility APIs, wireless printing capability, and a new experimental runtime virtual machine, called ART, which may come to replace Dalvik. KitKat debuted on Google's own Nexus 5 smartphone on October 31. 2013.

Overview

With all of these versions of Android, the features and changes to the OS have led to a rich and user-centered experience. The average user, who knows little to nothing about the technical aspects of the device, can operate the device like it is second nature. Now that you have acquired more insight into the underlying systems and architectures that make this possible, the only thing left to ask is, what's next?