CHAPTER 3
ANDROID OPEN SOURCE PROJECT
3.1. Introduction

With the revolution in mobile industries many mobile companies started competing to get maximum share in the market. Some of the biggest competitors in market before few years were Apple (IOS), RIM (Blackberry) and Nokia (Symbian and Later Nokia OS). Microsoft also had some share in the market with its Pocket PCs but not was very much in competition with other three. [26]

It was time when Applet dominated the market with highest number of users for their Iphone and Ipad devices. Almost equal number of Nokia and Blackberry devices were there in the market. Each had its unique reason for being sold but the common problem among them with was related to their code. These companies were keeping their source proprietary and were not sharing with the world.

In 2008 smart phones with a new operating system called Android came in market and people were curious to see it as the operating system was from Google Inc. If we look at the history of Android then a company called Android Inc. was founded in 2003 by Andy Rubin and his team to develop “smarter mobile devices that are more aware of its owner’s location and preferences” - Andy Rubin.

Google acquired Android Inc. on August 17, 2005 by giving an intimation to the mobile phone world that Google is now coming in this domain.

Later in 2007 OHA – Open Handset Alliance[32] was Formed, and in 2007 first Android Beta SDK was released. Company released Android 1.0 in 2008 for HTC’s phone called G1. That was the first step of Android in the mobile phone market and with the growing time it became number 1 and currently it has highest share of the mobile phone market.

After the launching Android has passed through many versions and variations and finally became a stable OS.
Some of the unique features in Android OS made it very popular in end users. It came with various attractions and different mobile device manufacturers like HTC, Samsung, etc... were ready to adopt these new inventions for their devices.

Since, Android was open source it became easier for manufacturers to design devices by customizing source based on their requirements. This not only made it possible for users to get android phones in affordable cost but also with some very fancy features also.

It is being now around 5 years since Android is in market and it is dominating mobile market from last 4 years. Following graph depicts growth and consistent dominance of Android over other competitors in last 4 years.

![Global Smartphone Sales - Gartner](https://example.com/graph.png)

**Figure 3.1** – Global Smart Phone Sales – Gartner

Above graph clearly indicates that users accepted Android based phones and continued using them for long time. If you see at the graph then it shows downfall of Blackberry, Microsfot and Symbian market (with some effect on iPhone)
As per Gartner's report published in May 2013 Android's share at the end of First Quarter of 2013 is 74.4%. Which is an increase of almost 18% compare to the first quarter sale of 2012 (56.9%)[31].

The data also tells about bad run for other big companies like Apple (iOS), Blackberry and Microsoft with others.

Except Iphone all other companies have less than 5% share of the smartphone market. This is the evidence that android is far popular and accepted worldwide by smartphone users.

Detailed figures related to “worldwide Smartphone Sales to End Users by Operating System in 1Q13 (Thousands of Units)” are given in the following table.

<table>
<thead>
<tr>
<th>Operating System</th>
<th>1Q13 Units Sold</th>
<th>1Q13 Market Share (%)</th>
<th>1Q12 Units Sold</th>
<th>1Q12 Market Share (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td>156,186.0</td>
<td>74.4</td>
<td>83,684.4</td>
<td>56.9</td>
</tr>
<tr>
<td>iOS</td>
<td>38,331.8</td>
<td>18.2</td>
<td>33,120.5</td>
<td>22.5</td>
</tr>
<tr>
<td>BlackBerry</td>
<td>6,218.6</td>
<td>3.0</td>
<td>9,939.3</td>
<td>6.8</td>
</tr>
<tr>
<td>Microsoft</td>
<td>5,989.2</td>
<td>2.9</td>
<td>2,722.5</td>
<td>1.9</td>
</tr>
<tr>
<td>Bada</td>
<td>1,370.8</td>
<td>0.7</td>
<td>3,843.7</td>
<td>2.6</td>
</tr>
<tr>
<td>Symbian</td>
<td>1,349.4</td>
<td>0.6</td>
<td>12,466.9</td>
<td>8.5</td>
</tr>
<tr>
<td>Others</td>
<td>600.3</td>
<td>0.3</td>
<td>1,242.9</td>
<td>0.8</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>-------</td>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>Total</td>
<td>210,046.0</td>
<td>100.0</td>
<td>147,020.2</td>
<td>100.0</td>
</tr>
</tbody>
</table>

Source: Gartner (May 2013)

The data is collected worldwide so it reflects acceptance of Android by people around the world. If it could be only from one region then it might have been ignored or might be if sale is good in one year but not other years then also it may not prove Android's popularity. But consistent growth and acceptance by worldwide users say that there is something different in Android.

As per Google, every day more than 1 million new Android devices are activated worldwide. Which confirms Gartner's report.

From above details many questions can be raised in one's mind:

1. Why Android is so popular?
2. It is younger than Apple, Microsoft, Blackberry and Symbian then also how it managed to get the popularity?
3. Why features provided by Android (or at least similar) are not offered by others?
4. Which are the factors that play important role in this huge sale all over the world?

If we try to find out answers for above questions then we will come to know that its the implementation, open source and huge number of applications for Android phones made it possible for Android. As being open source based on Linux Kernel, Android has many advantages some of them are listed below:
1. Any device manufacturer can use Android OS code and customize it as per its requirement.
2. Device manufacturers can sale their devices at lower price since cost of the OS is reduced.
3. Manufacturers compete with each other in healthy competition to serve the customers in better way.
4. Users get benefit of the healthy competition between manufacturers by getting various advanced features, schemes and attractions.

Android based devices range between ~3000 INR to ~40000 INR and these range can target users from all the classes of the society. The devices coming in the lowest prize are also far better with the other company's devices as they are developed on an open source Operating System (Android) and come with basic features.

One distinct feature for android users compared to other competitors is availability of huge number of applications from Google Play. IOS and Blackberry also has a place from where users can download applications for their devices but they are not able to attract developers and companies for building applications to support their devices. While android has motivated developers by keeping source code open and also by allowing any registered users to upload the application.

Definitely there are strict rules for publishing applications on IOS market (who is in direct competition with android) but it has allowed them to keep their market clean and safe. While android market does not have any mechanism to filter out applications based on content and features. This freedom allows developers to upload and publish any type of content in the form of application to Android market. This applications may sometime contain malware in them.

There is one more issue in android market related to duplicate applications. You can find multiple applications having same features/functionalities on android market which not only increases number of applications (rather duplicate applications) but also confuses users in selecting the right one.
As per Google's official data there are around 1.5 billion downloads of applications from Google play by android users.

There are millions of applications on Google play (Official android market) and out of them more than half are free applications so any android phone user can download it without paying any amount. Availability of a big number of applications on market keeps android users always busy in experimenting or using different applications. It also is a good revenue generator for android applications development firms.

There are many other reasons behind success of Android, but in simple words if we conclude shortly then we can say that Android dominates the Smart phone market and as per the predictions it will remain at the top for long time.

**AOSP**

The Android Open Source Project (AOSP)\(^{33}\) is an initiative created to guide development of the Android mobile platform. The Android platform consists of the operating system (OS), middleware and integral mobile applications.

The AOSP is overseen by the Open Handset Alliance (OHA), a Google-led coalition of over 30 wireless carriers, semiconductor companies, handset manufacturers and software companies. The purpose of the OHA is to further the development of open source standards for mobile devices.

The OHA member list includes Qualcomm, Broadcom, HTC, Intel, Samsung, Motorola, Sprint, Texas Instruments and Japanese wireless carriers KDDI and NTT DoCoMo. Companies that are conspicuously absent from the OHA member list include Nokia, Symbian, Apple, RIM, Microsoft, Verizon and Cingular.

In other words, Android is an open-source software stack created for a wide array of devices with different form factors. The primary purpose of Android is to create an open software platform available for carriers, OEMs, and developers to make their
innovative ideas a reality and to create a successful, real-world product that improves the mobile experience for end users. Google also wanted to make sure that there was no central point of failure, where one industry player could restrict or control the innovations of any other. The result is a full, production-quality consumer product whose source is open for customization and porting.
3.2. Android Architecture

After getting basic idea about popularity of Android and its market share, it is now time to understand how android is designed technically and what is Android Open Source Project (AOSP).

People commonly know android as an Operating System, but it is not only Operating System, it is a software stack and android architecture is technically known as Android Software Stack.

**Android Architecture or Android Software Stack**

![Android Software Stack / Architecture](image)

**Figure 3.2** – Android Software Stack / Architecture
**Linux Kernel**

Core services (including hardware drivers, process and memory management, security, network, and power management) are handled by a Linux 2.6 kernel. The kernel also provides an abstraction layer between the hardware and the remainder of the stack.

**Libraries**

Running on top of the kernel, Android includes various C/C++ core libraries such as libc and SSL, as well as:

- A Surface manager to provide display management
- Graphics libraries that include SGL and OpenGL for 2D and 3D graphics
- SQLite for native database support
- A media library for playback of audio and video media
- SSL and WebKit for integrated web browser and Internet security

**Android Run Time**

What makes an Android phone an Android phone rather than a mobile?

Linux implementation is the Android run time. Including the core libraries and the Dalvik virtual machine, the Android run time is the engine that powers your applications and, along with the libraries, forms the basis for the application framework.

- **Core Libraries** While Android development is done in Java, Dalvik is not a Java VM. The core Android libraries provide most of the functionality available in the core Java libraries as well as the Android-specific libraries.
- **Dalvik Virtual Machine** Dalvik is a register-based virtual machine that’s been optimized to ensure that a device can run multiple instances efficiently. It relies on the Linux kernel for threading and low-level memory management.

67
Application Framework

The application framework provides the classes used to create Android applications. It also provides a generic abstraction for hardware access and manages the user interface and application resources.

Application Layer

All applications, both native and third party, are built on the application layer using the same API libraries. The application layer runs within the Android run time using the classes and services made available from the application framework.

Above layers are the main layers of android software stack and so called architecture. Some of the important components of the architecture are discussed below:

The Dalvik Virtual Machine

One of the key elements of Android is the Dalvik Virtual Machine. Rather than using a traditional Java Virtual Machine (VM) such as Java ME (Java Mobile Edition) does, Android uses its own custom VM designed to ensure that multiple instances run efficiently on a single device.

The Dalvik VM uses the device’s underlying Linux kernel to handle low-level functionality including security, threading, and process and memory management. It is also possible to write C/C++ applications that run directly on the underlying Linux OS. While you can do this, in most cases there is no reason you should need to.

All Android hardware and system service access is managed using Dalvik as a middle tier. By using a VM to host application execution, developers have an abstraction layer that ensures they never have to worry about a particular hardware implementation.
The Dalvik VM executes Dalvik executable files (.dex), a format optimized to ensure minimal memory footprint. The .dex executables are created by transforming Java language compiled classes using the tools supplied within the SDK.

**Android Application Architecture**

Android’s architecture encourages the concept of component reuse, allowing you to publish and share activities, services, and data with other applications with access managed by the security restrictions you put in place.

The same mechanism that lets you produce a replacement contact manager or phone dialer can let you expose your application components to let other developers create new UI front ends and functionality extensions, or otherwise build on them.

The following application services are the architectural cornerstones of all Android applications, providing the framework.

- **Activity Manager**: Controls the life cycle of your activities, including management of the activity stack.
- **Views**: Are used to construct the user interfaces for your activities
- **Notification Manager**: Provides a consistent and non-intrusive mechanism for signaling users.
- **Content Providers**: Lets your applications share data between applications
- **Resource Manager**: Supports non-code resources like strings and graphics to be externalized

**Android Libraries**

Android offers a number of APIs for developing your applications. The following list of core APIs should provide an insight into what’s available; all Android devices will offer
support for at least these APIs:

- **android.util**: The core utility package contains low-level classes like specialized containers, string formatters, and XML parsing utilities.

- **android.os**: The operating system package provides access to basic operating system services like message passing, interprocess communication, clock functions, and debugging.

- **android.graphics**: The graphics API supplies the low-level graphics classes that support canvases, colors, and drawing primitives, and lets you draw on canvases.

- **android.text**: The text processing tools for displaying and parsing text.

- **android.database**: Supplies the low-level classes required for handling cursors when working with databases.

- **android.content**: The content API is used to manage data access and publishing by providing services for dealing with resources, content providers, and packages.

- **android.view**: Views are the core user interface class. All user interface elements are constructed using a series of Views to provide the user interaction components.

- **android.widget**: Built on the View package, the widget classes are the “here’s one we created earlier” user-interface elements for you to use in your applications. They include lists, buttons, and layouts.
• **com.google.android.maps**: A high-level API that provides access to native map controls that you can use within your application. Includes the MapView control as well as the Overlay and MapController classes used to annotate and control your embedded maps.

• **android.app**: A high-level package that provides access to the application model. The application package includes the Activity and Service APIs that form the basis for all your Android applications.

• **android.provider**: To ease developer access to certain standard Content Providers (such as the contacts database), the Provider package offers classes to provide access to standard databases included in all Android distributions.

• **android.telephony**: The telephony APIs give you the ability to directly interact with the device’s phone stack, letting you make, receive, and monitor phone calls, phone status, and SMS messages.

• **android.webkit**: The WebKit package features APIs for working with Web-based content, including a WebView control for embedding browsers in activities and a cookie manager.

In addition to the Android APIs, the Android stack includes a set of C/C++ libraries that are exposed through the application framework.

These libraries include:

• **OpenGL**: The library used to support 3D graphics based on the Open GL ES 1.0 API
• **FreeType**: Support for bitmap and vector font rendering

• **SGL**: The core library used to provide a 2D graphics engine

• **libc**: The standard C library optimized for Linux-based embedded devices

• **SQLite**: The lightweight relation database engine used to store application data

• **SSL**: Support for using the Secure Sockets Layer cryptographic protocol for secure Internet communications

**Advanced Android Libraries**

The core libraries provide all the functionality you need to start creating applications for Android, but it won’t be long before you’re ready to delve into the advanced APIs that offer the really exciting functionality.

Android hopes to target a wide range of mobile hardware, so be aware that the suitability and implementation of the following APIs will vary depending on the device upon which they are implemented.

• **android.media**: The media APIs provide support for playback and recording of audio and video media files, including streamed media.

• **android.opengl**: Android offers a powerful 3D rendering engine using the OpenGLES API that you can use to create dynamic 3D user interfaces for your applications.

• **android.hardware**: Where available, the hardware API exposes sensor hardware including the camera, accelerometer, and compass sensors.
- **android.location**: The location-based services API gives your applications access to the device’s current physical location. Location-based services provide generic access to location information using whatever position-fixing hardware or technology is available on the device.

- **android.bluetooth, android.net.wifi, and android.telephony**: Android also provides low-level access to the hardware platform, including Bluetooth, Wi-Fi, and telephony hardware.
3.3. Why Android?

Following are some of the reasons why I selected Android for my research work.

**Android is Open Source**

As discussed in last section, Android is an open source project controlled by OHA and complete Android source is available for developers, manufacturers and researchers to study and customize. Of course there are some other open source alternatives available but android has many other advantages like biggest user base and largest share in smart phone market.

**Android is Java Based**

Another advantage of android is, its source code is Java based and since Java is stable and secure it is expected that Android will also remain stable and secure. So the advantages of Java are always with Android OS.

**Popularity of Android**

As discussed in last section, android has largest market share in smart phone market and is the most popular OS at this time. Selection of such a popular OS which is used by most number of users can benefit maximum number of end users.

**Future of Android**

Android is currently at top of the market and it is predicted by the analysts that it will continue dominating market for at least next decade or so.