

Chapter One

Introduction

What does mobile phone programming encapsulate?



User-interface design



Operating system design



programming languages



networking



cloud computing

2

Mobile phones got their name from their very first application, **allowing people to make phone calls while being mobile**. The communication architecture was dominated by *base stations communicating with the mobile phones*. Base stations were needed to enter the existing telephone networks and thereby allowing communicating with fixed line communication partners. Furthermore, the base stations and the connected backbone allowed communicating with other mobile users in different cells. In the beginning, referred to as *first generation (1G)* of mobile networks, there was no need for additional functionality, **besides voice services, on top of the mobile phones**. 1G refers to the first generation of wireless telephone technology, mobile telecommunications which was first introduced in 1980s and completed in early 1990s.

- Its Speed was up to 2.4kbps.
- It allows the calls in 1 country. (no roaming)
- It allows the voice calls Only
- 1G network use Analog Signal.

Drawbacks of 1G

- Poor Voice Quality
- Poor Battery Life
- Large Phone Size
- No Security
- Limited Capacity
- Poor Handoff Reliability

With the *second generation (2G)*, changing from analog mobile systems to digital ones, enhanced services were introduced such as the **short message service (SMS)** and **data connections to the Internet**. But another important change took place: the split of *network and service providers* such that we have four major players in the mobile world as given in Figure 1.1. From that time on the mobile market has been divided into four different players, namely **the customer** as the main target of the business, **the network providers** enabling to convey bits toward the customer, **the service provider**, and **the mobile phone manufacturer**. These four entities have some interdependencies.

The most dominating one is the relationship between network providers and mobile phone manufacturers. To attract customers, the network providers bundle their contracts with a mobile phone, both being renewed within a given period. This ensures the mobile phone Manufacturers sell a huge number of mobile phones.

On the other side, the network operators also pull strings as to which services and applications are available on the mobile phone. Furthermore the network operator also controls which services are going over their network and makes the service provider dependent on the network operators' decision. This relationship should be taken into account making novel applications for mobile phones.

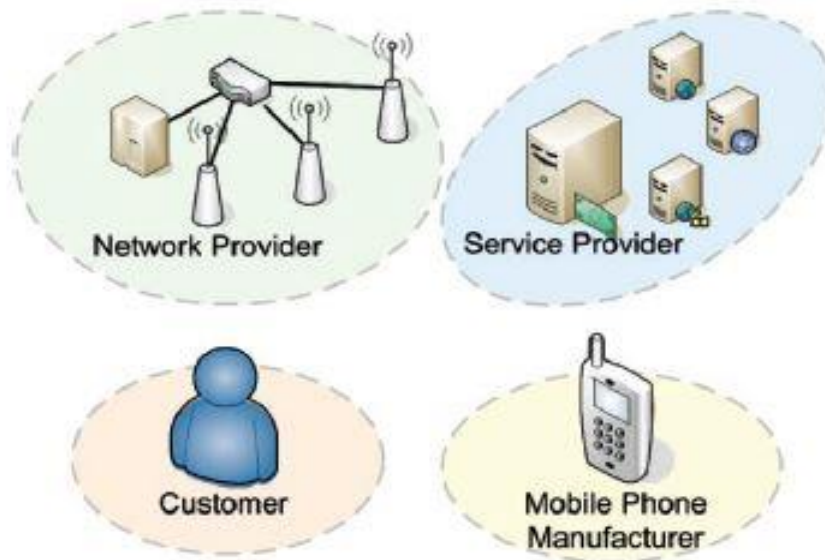


Figure 1.1. Interdependencies between the mobile phone customer, the network provider, the service provider, and the mobile phone manufacturer.

2G technology refers to the 2nd generation which is based on GSM.

- It was launched in Finland in the year 1991.
- 2G network use digital signals.
- Its data speed was up to 64kbps.

Features Includes:

- It enables services such as text messages, picture messages and MMS (multimedia message).
- It provides better quality and capacity.

Drawbacks of 2G

- 2G requires strong digital signals to help mobile phones work. If there is no network coverage in any specific area, digital signals would weak. (strong network coverage)
- These systems are unable to handle complex data such as Videos.

2.5 G Technology

2.5G sometimes called 2G Cellular is a technology between the second (2G) and third (3G) generation of mobile telephony.

Features Includes:

- ✓ **Phone Calls**
- ✓ **Send/Receive E-mail Messages**
- ✓ **Web Browsing**
- ✓ **Speed : 64-144 kbps**
- ✓ **Camera Phones**
- ✓ **Take a time of 6-9 mins. to download a 3 mins. Mp3 song.**

The *third generation* (3G) of mobile networks refers to third generation which was introduced in year 2000s. **Data Transmission speed increased from 144kbps- 20Mbps.** typically called Smart Phones and **features increased its bandwidth and data transfer rates to accommodate web-based applications and audio and video files.**

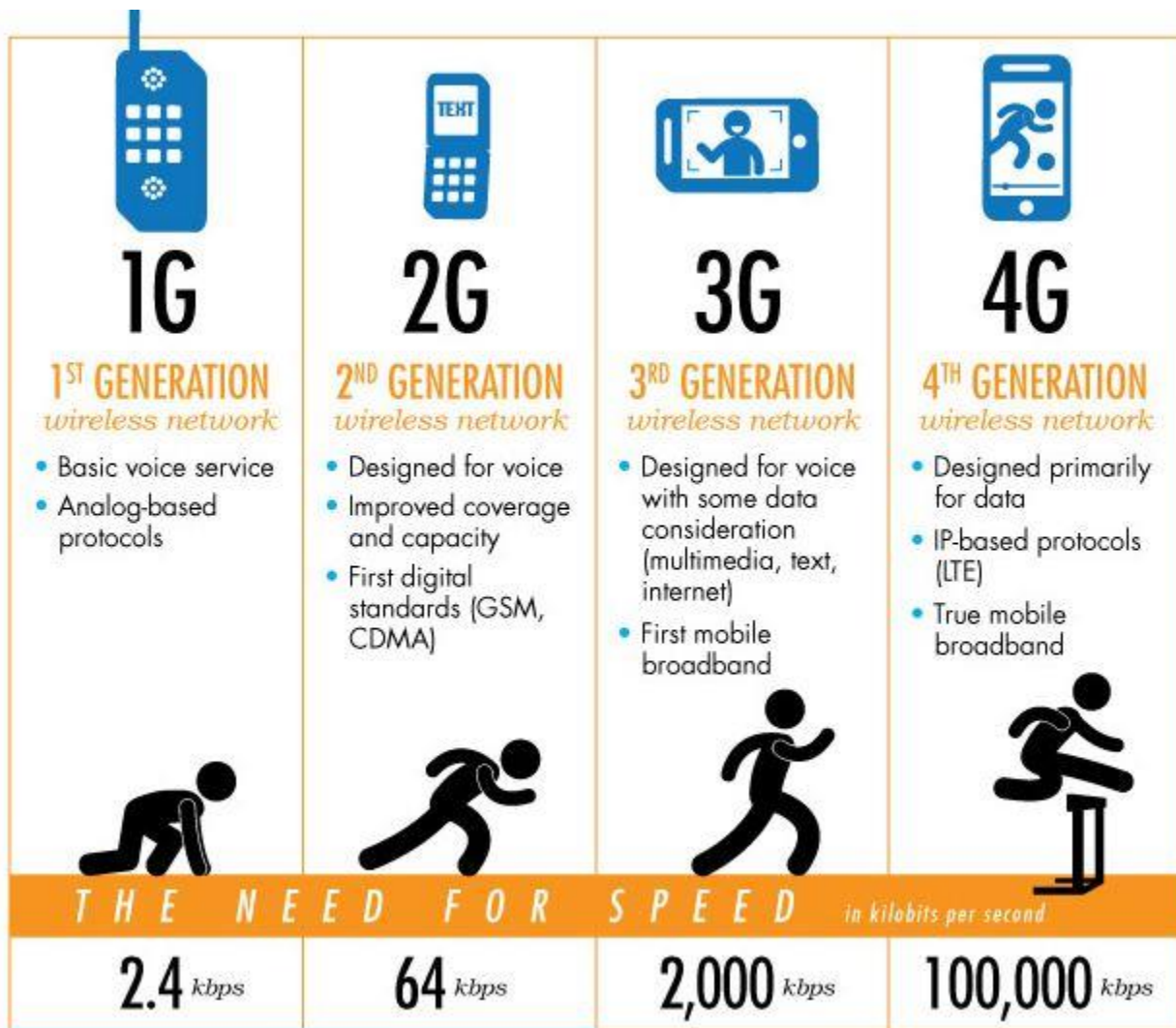
Features of 3g Technology

- ❖ **Video Calls**
- ❖ **Better Call Quality**
- ❖ **Expect Low Price Calls (ASIACELL?!)**
- ❖ **Providing Faster Communication**
- ❖ **(Better Upload & Download)**
- ❖ **Providing Good Wireless Internet**
- ❖ **Send/Receive Large Email Messages**
- ❖ **High Speed Web / More Security**
- ❖ **Good Marketing and Business**
- ❖ **Video Conferencing / 3D Gaming**
- ❖ **TV Streaming/ Mobile TV/ Phone Calls**
- ❖ **Large Capacities and Broadband Capabilities (extra millions of subscribers)**
- ❖ **11 sec – 1.5 min. time to download a 3 min Mp3 song.**

Drawbacks of 3G Technology

- **Expensive fees for 3G Licenses Services (more than 1 billion \$)**

- It was challenge to build the infrastructure for 3G (ASIACELL Changed 550 Towers)
- High Bandwidth Requirement (More Frequency Bands 2100 MHz)
- Expensive 3G Phones.
- Large Cell Phones.



4G Technology (Anytime, Anywhere)

4G technologies refer to or short name of fourth Generation which was started from late 2000s. (4G/LTE), Capable of providing 100Mbps – 1Gbps speed. One of the basic terms used to describe 4G is **MAGIC**.

MAGIC:

- ❖ **Mobile Multimedia**
- ❖ **Anytime Anywhere**
- ❖ **Global Mobility Support**
- ❖ **Integrated Wireless Solution**
- ❖ **Customized Personal Services**

Also known as Mobile Broadband Everywhere

- The next generations of wireless technology that promises **higher data rates and expanded multimedia services.**
- Capable to provide speed 100Mbps-1Gbps.
- High QOS and High Security
- Provide any kind of service at any time as per user requirements, anywhere.

Features Include:

- **More Security**
- **High Speed**
- **High Capacity**
- **Low Cost Per-bit etc.**

Drawbacks of 4G

- ✓ **Battery uses is more**
- ✓ **Hard to implement**
- ✓ **Need complicated hardware**
- ✓ **Expensive equipment required to implement next generation network.**

5G Technology

5G technology refer to short name of fifth Generation which was started from late 2010s. Complete wireless communication with almost no limitations. It is highly supportable to WWW(Wireless World Wide Web).

Benefits of 5G Technology

- ❖ **High Speed, High Capacity**
- ❖ **5G technology providing large broadcasting of data in Gbps .**
- ❖ **Multi - Media Newspapers, watch T.V programs with the clarity as to that of an HD Quality.**
- ❖ **Faster data transmission that of the previous generations. Large Phone Memory, Dialing Speed, clarity in Audio/Video.**
- ❖ **Support interactive multimedia, voice, streaming video, Internet and other.**
- ❖ **5G is More Effective and More Attractive.**



The personal and the community services are separated into services with and without wireless network support. The wireless network support is realized by standard GSM connection, GPRS, EDGE, 3G data connections, Bluetooth, or WLAN. **The personal services without network support include services such as calendars, cameras, downloading facilities to change the logo on the phone, and single-player games such as Tetris, pong, snake,**

etc. The logo services gave the network and service providers big revenues as they enabled people to customize their mobile phones.

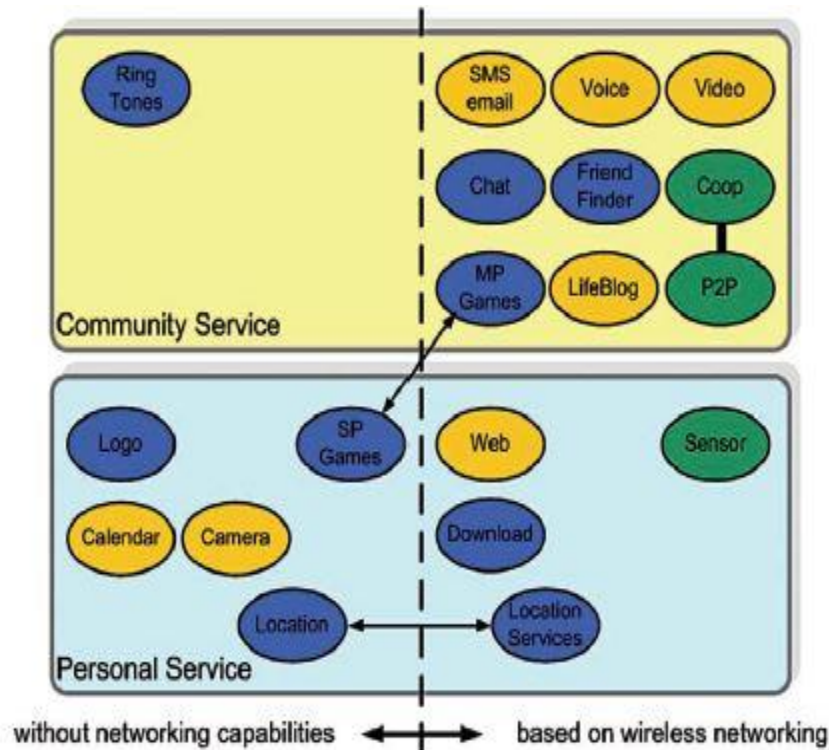


Figure 1.2. Classification of services into service and community services.

1.2 The Flexible Mobile Phone

The mobile phone platform has become more and more flexible for third party developers. The first programming language on the mobile phone was JAVA and now there are several ways to implement an application on the phone.

As given in Figure 1.3 the applications run in the *application suite*. The *user interface frameworks* host all functionality that let us operate the mobile device. Qtopia is one example of that. The mobile phone manufacturer offers access to their platforms (with some restriction) using their kernel and hardware drivers such as the Symbian OS to access the hardware platform. So nearly anything that is in software can be changed on the mobile.

The only static part of the phone is the hardware platform. Here and there some APIs can be found to change some parameters for a given wireless technology, but we are far from a composable or software-defined radio.

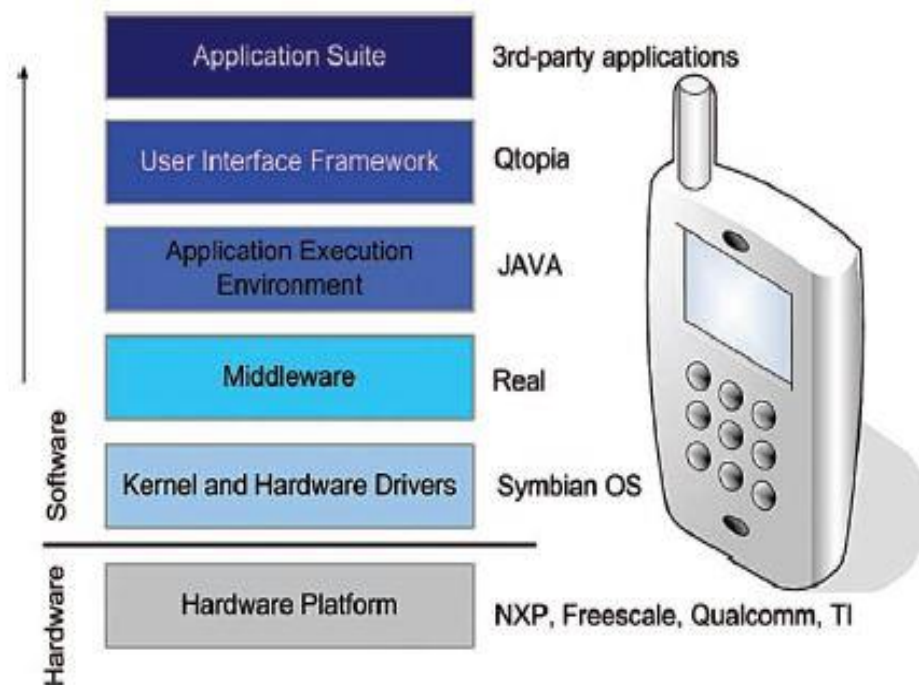


Figure 1.3. General architecture of the programmable mobile phone.

More flexibility on the phone as given in Figure a mobile device has many different capabilities and functionalities. We have grouped those of a mobile (smart) phone into three classes, namely user interfaces, communication interfaces, and built-in resources.

The user interfaces comprise typically the **speaker, microphone, camera, display, built-in sensors, and keyboard capabilities**. **The built-in terminal resources** include the **battery, the central processing unit, and the data storage**. From our point of view we highlight particularly the **communication interfaces**, which typically include **cellular and short-range capabilities**.



Figure 1.4. A mobile device broken up into the several capabilities grouped into the user interface, communication interface, and the built-in resources.

1.3 Wireless Technologies and Architectures

The wireless network capabilities in mobile phones are the main enablers to invent new appealing applications. The technologies differ in **the supported data rates, the communication protocols used, the communication coverage, and the energy consumption related to it**. In general we distinguish between short-range and cellular communication systems.

1.3.1 Cellular Communication Systems

In the beginning the main service on mobile phones was voice. The first logical move was to enable connections of mobile users with users connected to fixed terminals or to use the core network to extend virtually a connection between two mobile users.

1.3.2 Short-Range Communication Systems

As candidates for the short-range communication we **refer to wireless personal area networks (WPAN) and wireless local area networks (WLAN)**. As a candidate for WPAN and WLAN, Bluetooth and the IEEE802.11 family is named here, respectively. **Bluetooth** was introduced by

Ericsson in 1999 as a cable replacement for mobile equipment to ease the data exchange between mobile phones, personal digital assistants, laptops, and personal computers.

Chapter Two

2.1 Service Discovery

Service discovery deals with locating services which can be used to accomplish a task. For example discovering a printer to print documents on, or locating a shop offering the service of hairdressing. When you locate a service you usually obtain an address which you use to invoke the offered service.

2.1.1 Service Discovery in Computer Networks

In networks which use some kind of shared media, such as Ethernet or wireless networks, each node is usually assigned an address and all transmitted packets which open up different communication methods, as shown in Figure 2.3:

Unicast: From one node to another node, as shown in Figure 2.3A where node A sends a packet to node B. In IPv4-networks this could be from source address 10.0.0.1 to destination address 10.0.0.5.

Broadcast: From one node to all other nodes in the network, as shown in Figure 2.3B where node A sends a packet to all the other nodes. In IPv4-networks this is usually done by sending to IPv4 address 255.255.255.255.

Multicast: One-to-many traffic has many use cases, but it is not always that you want "many" to be the whole network. With multicasting a node must join a multicast-group before it retrieves packets, as shown in Figure 2.3C where nodes A, B, C, and D are all part of one multicast group. Node A is here sending a packet which all the group members receive. For IPv4 networks the address range 224.0.0.0 through 239.255.255.255 has been reserved for multicasting.

Anycast: This is multicasting with a twist: the packet is only delivered to the nearest member of the group. This routing scheme is usually used for load balancing.

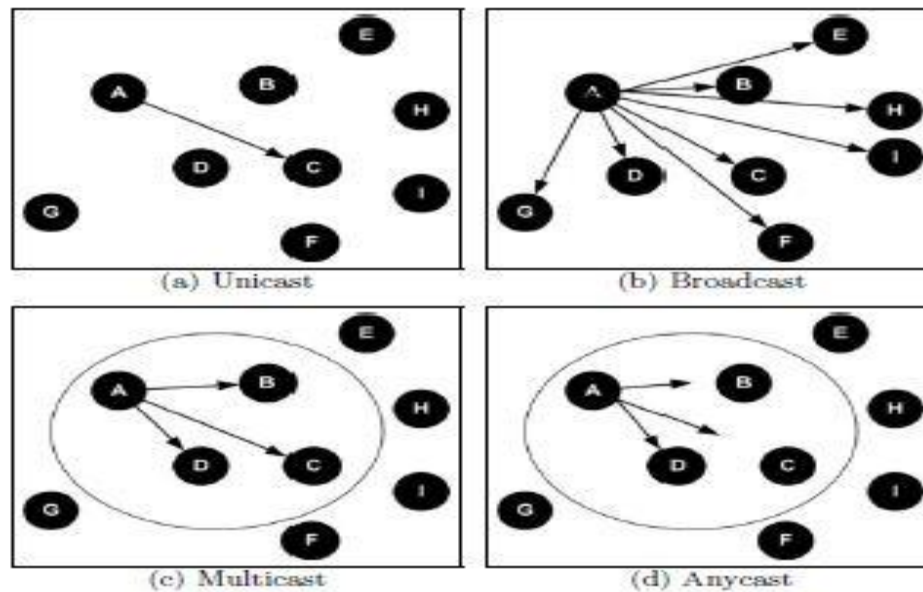


Figure 9.1. Different routing schemes.

There are three different ways to provide nodes with these addresses:

Manual configuration, dynamic configuration, and random configuration.

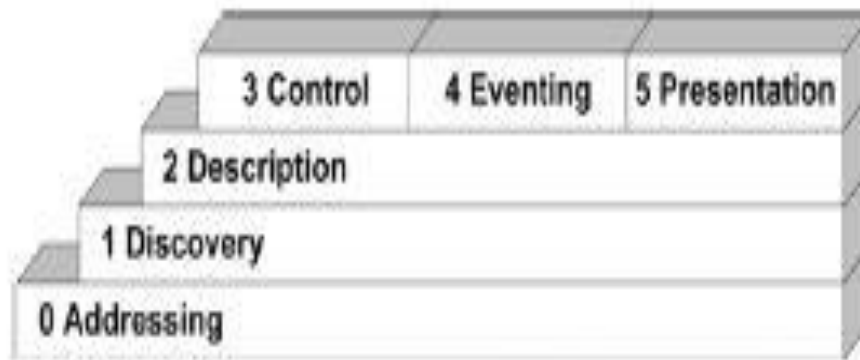
With manual configuration, or static configuration, **an administrator must manually configure each node in the network**. In dynamic configuration **there is a central service which hands out addresses to nodes**. Random configuration makes it easier. In IPv4 networks a special range, 169.254/16, has been reserved for this purpose. The host should first check for any available DHCP server, and if none is available then randomly create an IP-address from that special range.

2.2 Universal Plug & Play

UPnP technology enables devices to connect together and form a pervasive peer-to-peer network where they can control each other. Such a distributed network is very powerful because it enables the devices to get more intelligent together.

2.2.1 UPnP Device Architecture

We will now go through the six phases of UDA, as shown in the Figure. Please note that we cannot cover all the details of the UDA here.



2.3 What is Mobile Device Management (MDM)?

Mobile device management (MDM) is a type of security software used by an IT department to monitor, manage and secure employee's mobile devices that are deployed across multiple mobile service providers and across multiple mobile operating systems being used in the organization. Mobile device management software is often combined with additional security services and tools to create a complete mobile device and security management solution.

The Gartner research firm defines mobile device management as "a range of products and services that enables organizations to deploy and support corporate applications to mobile devices, such as smartphones and tablets, possibly for personal use — enforcing policies and maintaining the desired level of IT control across multiple platforms."

2.4 Mobile Device Management Solutions

Most mobile device management solutions **provide organizations with end-to-end security — meaning the mobile apps, network and data used by the mobile device (in addition to the mobile device itself) are managed by an organization's IT department with a single mobile device software product.**

Some enterprise MDM solutions combine mobile security and expense management in a single product. Depending on the vendor and what specific features it supports, you can typically expect mobile device management software to contain some or all of the following features:

management and support of mobile applications, mobile policy management, inventory management, security management and telecom service management.

The most common operating systems used in Smartphone are Linux, Windows Mobile from Microsoft, Symbian, RIM BlackBerry, Android, and Mac OSX.

2.5 Mobile phone Environment

Java technology is an object-oriented programming environment created by **Sun Microsystems**. **The main goal** was to offer an environment to develop platform-independent applications following the slogan *write once, run anywhere*. Today Java is one of the most popular software developing environments. It offers the opportunity to create enterprise applications for server and desktop computers as well as applications for small and mobile devices. Java has been split up in 4 distinct editions 3:

- ❖ **J2EE** is mainly intended for building distributed enterprise applications with emphasis on server-side development and web applications.
- ❖ **J2SE** is intended for conventional desktop applications development.
- ❖ **J2ME** is a subset of J2SE intended for embedded devices which cannot support a full J2SE implementation.
- ❖ **Java Card** offers an environment for developing applications running on smart cards.



Figure 3.1. Java family architecture.

2.6 J2ME Platform

J2ME is part of the Java family intended to build applications running on battery driven platforms such as mobile phones, PDAs, and other embedded devices. All these devices have different capabilities such as **user interface, memory budget, and processing power**. At the moment two configurations exists in the J2ME platform:

- **CLDC**, for devices with:
 - Very simple User Interface
 - Low level memory budget (160 Kb to 512 Kb) for Java
 - Wireless communication, targeting low bandwidth
 - 16 or 32 bit processors
 - Limited power, often battery operating
 - Examples: mobile phones, pagers, entry level PDAs.

These devices will here be known as *less capable devices*.

- **CDC**, for devices with:
 - Large range of User Interface capabilities
 - Memory budget in the range of 2–16 MB for Java
 - Connectivity to some type of network
 - 16- or 32- bit processors
 - Examples: Internet TVs, TV set-top boxes, PDA.

The J2ME platform runs on the top of the operating system of the Java-enabled mobile phones and consists of two different versions: one for ***more capable devices*** and one for ***less capable devices***.

Each of them includes:

Configuration: This is the basic set of class libraries and the VM which interprets and runs the applications. The VM is either a full JVM or a subset of this. For CLDC and CDC the class libraries are mostly a subset of the J2SE API in addition to APIs specific for J2ME such as the GCF.

Profiles: The profiles complement the configuration with more high level APIs specific for a device group. **APIs provided** by a profile normally **consists of application life-cycle model,**

user interface, storage, specific network support, and more. Notice that profiles can be layered on top of each other, as will be seen with CDC. The configuration together with one or more profiles creates a complete J2ME runtime environment.

Optional Packages: With optional packages the runtime environment is extended with APIs targeting specific requirements, **such as database connectivity, communication methods, graphics,** etc. An optional package can be configuration independent (i.e., support both CLDC and CDC) or it can be targeted at one specific configuration.

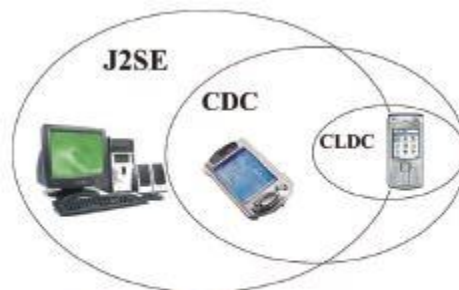


Figure 3.2. J2ME inheritance.

2.6.1 Less Capable Devices: Targeting CLDC

As stated earlier, J2ME can target two different groups of devices. In this section the configuration and the profile supported by *less capable devices* are described.

Connected Limited Device Configuration

CLDC defines a set of libraries and a VM for resource constrained devices. At the moment two versions of CLDC are available:

- **CLDC 1.0 is the first release of the CLDC specification.**
- **CLDC 1.1 is a backward compatible version of the CLDC 1.0 with some enhancements, such as support for floating-point math and weak references.**

2.7 Development Environment

For J2ME development you need a Java compiler, a J2ME runtime and an editor to write code in. This is called the development environment and is explained here starting with the lowest

layer and moving upward the tool chain. The development environment explained in the following sections will be based on this:

- **Java SE Development Kit (JDK)**
- **NetBeans IDE**
- **NetBeans Mobility Pack for CLDC and/or CDC**

2.8 What Are the Different Types of Mobile Phone Operating Systems?

There are many different mobile phone operating systems currently being used. Some of the most well-known include Symbian, BlackBerry OS, iOS, Windows Mobile, and Android. Most of them are associated with specific brands of phone made by specific companies while others are open-source and are available on a variety of platforms.

Symbian is primarily found on Nokia phones, but phones by Sharp, Fujitsu, Sony and other companies have the operating system installed as well. **Symbian is open source, meaning anyone can use it without having to pay**. It is widely used, but it is not the most advanced or full-featured of mobile phone operating systems. Most phones that use Symbian **are low-end devices, not full-featured smart phones**.

BlackBerry OS. It is the only one of the mobile phone operating systems to be found on BlackBerry devices. The BlackBerry OS is tailor-made for business, with functionality taking a seat over personality or appearance. **Its main focus is on messaging, email and other communication features. Media players and other entertainment-based applications are less-common for the device.**

Apple's iOS is found on the iPhone, iPad and iPod Touch. Operating system security and compatibility has been a point of contention for some in regards to iOS. It is entirely closed-source, and Apple chooses on its own which software the platform will and will not support. Adobe Flash for example.

Windows Mobile, also known as Windows Phone, is the mobile phone version of Microsoft's operating system. Because of this, it is easily compatible with many Windows programs such as **Microsoft Office**, making it a popular choice for businesspeople.

The Android is based off of the Linux operating system. Originally developed by an independent organization, it was later bought by Google, although the OS itself remains free and open source. Android is praised by many for his flexibility as a platform. Anyone can develop apps for the OS, and any company can release a phone using it.

2.9 Mobile Device Management Algorithms

2.9.1 First-In, First Choice FIFC

First In- First Choice – is a service policy whereby the requests of customers or clients are attended to in the order that they arrived, without other biases or preferences. The policy can be employed when processing sales orders, in determining restaurant seating, on a taxi stand, etc. In Western society, it is the standard policy for the processing of most queues in which people wait for a service that was not prearranged or preplanned.

FIFC is an operating system process scheduling algorithm and a network routing management mechanism that automatically executes queued requests and processes by the order of their arrival. With first come, first served, what comes first is handled first; the next request in line will be executed once the one before it is complete.

Example1/ Consider the following set of processes assumed to have arrived at different times & orders with length of the Mobile Device Management. By using FIFC Algorithm

Actions	Size	Arrival Time
A	22	0
B	3	0
C	3	0

1. Draw the **Gantt chart** to show the execution of each process.
2. Calculate the **A.W.T** of each process.
3. Calculate the **T.A.T** of each process.

Example2/ Consider the following set of processes assumed to have arrived at different times & orders with length of the Mobile Device Management. By using FIFC Algorithm

Actions	Size	Arrival Time
A	7	2
B	1	0
C	2	1
D	1	2
E	5	3

1. Draw the **Gantt chart** to show the execution of each process.
2. Calculate the **A.W.T** of each process.
3. Calculate the **T.A.T** of each process.

Example3/ Consider the following set of processes assumed to have arrived at different times & orders with length of the Mobile Device Management. By using FIFC Algorithm

Actions	Size	Arrival Time
A	11	3
B	2	0
C	5	1
D	2	6
E	6	7

1. Draw the **Gantt chart** to show the execution of each process.
2. Calculate the **A.W.T** of each process.
3. Calculate the **T.A.T** of each process.

2.9.2 Best Algorithm

In computer science/data structures, a priority queue is an abstract data type which is like a regular queue or stack data structure, but where additionally each element has a "priority" associated with it. In a priority queue, an element with high priority is served before an element with low priority. If two elements have the same priority, they are served according to their order

in the queue. While priority queues are often implemented with heaps, they are conceptually distinct from heaps. A priority queue is an abstract concept like "a list" or "a map"; just as a list can be implemented with a linked list or an array, a priority queue can be implemented with a heap or a variety of other methods such as an unordered array.

Example1/ Consider the following set of processes assumed to have arrived at different times &orders with length of the Mobile Device Management. By using Best Algorithm

Actions	Arrival Time	Size	Best
A	0	6	8
B	0	2	6
C	0	16	5
D	0	8	7
E	0	11	3
F	0	6	2

1. Draw the **Gantt chart** to show the execution of each process.
2. Calculate the **A.W.T** of each process.
3. Calculate the **T.A.T** of each process.

Example2/ Consider the following set of processes assumed to have arrived at different times &orders with length of the Mobile Device Management. By using Best Algorithm

Actions	Arrival Time	Size	Best
A	0	10	6
B	5	2	2
C	9	7	4
D	10	9	8
E	15	7	3

1. Draw the **Gantt chart** to show the execution of each process.
2. Calculate the **A.W.T** of each process.
3. Calculate the **T.A.T** of each process.

Example3/ Consider the following set of processes assumed to have arrived at different times & orders with length of the Mobile Device Management. By using Best Algorithm

Actions	Arrival Time	Size	Best
A	0	6	8
B	5	2	6
C	8	16	5
D	9	8	7
E	12	11	3
F	14	6	2

1. Draw the **Gantt chart** to show the execution of each process.
2. Calculate the **A.W.T** of each process.
3. Calculate the **T.A.T** of each process.

Mobile Operating System (Mobile OS)

Much like the Linux or Windows operating system controls your desktop or laptop computer, a mobile operating system is the software platform on top of which other programs can run on mobile devices.

The operating system is responsible for determining the functions and features available on your device, such as thumbwheel, keyboards, WAP, synchronization with applications, email, text messaging and more. The mobile OS will also determine which third-party applications (mobile apps) can be used on your device.

1. Android OS (Google Inc.)

The Android mobile operating system is Google's open and free software stack that includes an operating system, middleware and also key applications for use on mobile devices, including smartphones. Updates for the **open source Android mobile operating system** have been developed under “dessert-inspired” codenames (Cupcake, Donut, Eclair, Gingerbread, Honeycomb, Ice Cream Sandwich) with each new version arriving in alphabetical order with new enhancements and improvements.

2. Bada (Samsung Electronics)

Bada is a proprietary Samsung mobile OS that was first launched in 2010. The Samsung Wave was the first smartphone to use this mobile OS. Bada provides mobile features such as **multipoint-touch, 3D graphics and of course, application downloads and installation.**

Bada was an operating system for mobile devices such as smartphones and tablet computers. It was developed by Samsung Electronics. Its name is derived from "바다 (bada)", meaning "ocean" or "sea" in Korean. It ranges from mid- to high-end smartphones. To foster adoption of Bada OS, since 2011 Samsung reportedly has considered releasing the source code under an open-source license, and expanding device support to include Smart TVs. All Bada-powered devices are branded under the Wave name, but not all of Samsung's Android-powered devices are branded under the name Galaxy.

3. BlackBerry OS (Research In Motion)

The BlackBerry OS is a proprietary mobile operating system developed by Research In Motion for use on the company's popular BlackBerry handheld devices. The BlackBerry platform is popular with corporate users as it offers synchronization with Microsoft Exchange, Lotus Domino, Novell GroupWise email and other business software, when used with the BlackBerry Enterprise Server.

4. iPhone OS / iOS (Apple)

Apple's iPhone OS was originally developed for use on its iPhone devices. Now, the mobile operating system is referred to as iOS and is supported on a number of Apple devices including the iPhone, iPad, iPad 2 and iPod Touch. The iOS mobile operating system is available only on Apple's own manufactured devices as the company does not license the OS for third-party hardware. Apple iOS is derived from Apple's Mac OS X operating system.

Editor's Pick: Need help understanding Apple mobile devices? Webopedia's "iPhone, iPhone 3G, and iPhone 3G S - What's The Difference?" article will help you get started.

5. MeeGo OS (Nokia and Intel)

A joint open source mobile operating system which is the result of merging two products based on open source technologies: Maemo (Nokia) and Moblin (Intel). MeeGo is a mobile OS designed to work on a number of devices including smartphones, netbooks, tablets, in-vehicle information systems and various devices using Intel Atom and ARMv7 architectures.

6. Palm OS (Garnet OS)

The Palm OS is a proprietary mobile operating system (PDA operating system) that was originally released in 1996 on the Pilot 1000 handheld. Newer versions of the Palm OS have added support for expansion ports, new processors, external memory cards, improved security and support for ARM processors and smartphones. Palm OS 5 was extended to provide support for a broad range of screen resolutions, wireless connections and enhanced multimedia capabilities and is called Garnet OS.

7. Symbian OS (Nokia)

Symbian is a mobile operating system (OS) targeted at mobile phones that offers a high-level of integration with communication and personal information management (PIM) functionality. Symbian OS combines middleware with wireless communications through an integrated mailbox and the integration of Java and PIM functionality (agenda and contacts). Nokia has made the Symbian platform available under an alternative, open and direct model, to work with some OEMs and the small community of platform development collaborators. Nokia does not maintain Symbian as an open source development project.

8. WebOS (Palm/HP)

WebOS is a mobile operating system that runs on the Linux kernel. WebOS was initially developed by Palm as the successor to its Palm OS mobile operating system. It is a proprietary Mobile OS which was eventually acquired by HP and now referred to as webOS (lower-case w) in HP literature. HP uses webOS in a number of devices including several smartphones and HP TouchPads. HP has pushed its webOS into the enterprise mobile market by focusing on improving security features and management with the release of webOS 3.x. HP has also announced plans for a version of webOS to run within the Microsoft Windows operating system and to be installed on all HP desktop and notebook computers in 2012.

9. Windows Mobile (Windows Phone 7)

Windows Mobile is Microsoft's mobile operating system used in smartphones and mobile devices – with or without touchscreens. The Mobile OS is based on the Windows CE 5.2 kernel. In 2010 Microsoft announced a new smartphone platform called Windows Phone 7.

Chapter Three

The mobile Operating System

Much like the Linux or Windows operating system controls your desktop or laptop computer, a mobile operating system is the software platform on top of which other programs can run on mobile devices. A mobile operating system, also called a mobile OS, is an operating system that is specifically designed to run on mobile devices such as mobile phones, smartphones, PDAs, tablet computers and other handheld devices.

3.1 What is Android?

Android is a Linux-based operating system designed primarily for touchscreen mobile devices such as smartphones and tablet computers, developed by Google in conjunction with the **Open Handset Alliance OHA**. Initially developed by Android Inc, whom **Google** financially backed and later purchased in 2005, Android was unveiled in 2007 along with the founding of the Open Handset Alliance, a consortium of 86 hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices.



Figure 1show android logo

The first Android-powered phone was sold in October 2008, and by the end of 2010 Android had become the world's leading smartphone platform. It had a worldwide smartphone market share of 68% at the second quarter of 2012, and as of Q3 2012, there were 500 million devices activated and 1.3 million activations per day. Operating Systems have developed a lot in last 15 years.

Android Inc was founded in Palo Alto of California, U.S. by Andy Rubin, Rich miner, Nick sears and Chris White in 2003. Later Android Inc. was acquired by Google in 2005. After original release there have been number of updates in the original version of Android.

Android 1.1 Feb 2009	<ul style="list-style-type: none">• Support for saving attachments for MMS• Marquee in layouts• API changes
Android 1.5 Cupcake April 2009	<ul style="list-style-type: none">• Bluetooth A2DP and AVRCP support• Uploading videos to YouTube and pictures to Picasa
Android 1.6 Donut Sep 2009	<ul style="list-style-type: none">• WGA screen resolution support• Google free turn by turn support
Android 2.0/1 Eclair Oct 2009	<ul style="list-style-type: none">• HTML5 file support• Microsoft exchange server• Bluetooth 2.1
Android 2.2 Froyo May 2010	<ul style="list-style-type: none">• USB tethering and Wi-Fi hotspot functionality• Adobe flash 10.1 support
Android 2.3 Gingerbread Dec 2010	<ul style="list-style-type: none">• Multi touch software keyboard• Support for Extra Large screen sizes and resolution
Android 3.0 Honeycomb May 2011	<ul style="list-style-type: none">• Optimized tablet support with a new user interface• 3D desktop• Video chat and Gtalk support

Figure 2 show android versions

Android is a powerful Operating System supporting a large number of applications in Smart Phones. These applications make life more comfortable and advanced for the users.

Hardware's that support Android is mainly based on ARM architecture platform. Some of the current features and specifications of android are:

Application framework	• It enables reuse and replacement of components
Dalvik virtual machine	• It is optimized for mobile device
Integrated Browser	• It is based on the open source Web kit engine
Optimized graphics	• It is powered by a custom 2D graphics library; 3D graphics based on the OpenGL ES 1.0 specification
SQLite	
Media support	
GSM Technology	
Bluetooth, EDGE, 3G, Wi-fi	
Camera, GPS, Compass etc	

Figure 3show current features and specifications of android

Android relies on Linux version 2.6 for core system services **such as security, memory management, process management, network stack, and driver model**. For software development, Android provides Android SDK (Software development kit).

3.2 Infrastructure for the operating system

Basically located the **Linux kernel**, its responsible for:

- **Display Driver**
- **Bluetooth Driver**
- **Camera Driver**
- **Flash Memory Driver**
- **Binder (IPC) Driver**
- **Keypad Driver**
- **USB Driver**
- **Wi-Fi Driver**
- **Audio Driver**
- **Power Management**

Then followed by a layer operating

- **Dalvik Virtual Machine**
- **Core Libraries**

These libraries are:

- **Surface Manager**
- **Media Framework**
- **SQLite**
- **OpenGL ES**
- **Free Type**
- **Web Kit**
- **SGL**

- SSL
- Libbc

Then followed this layer is **application layer**, which is used by all the applications

- Activity Manager
- Window Manager
- Content Provider
- View System
- Notification Manager
- Package Manager
- Telephony Manager
- Resource
- Location Manager
- Sensor Manager

Then in the last layer comes the main applications downloaded by the company such as:

- Home
- Contacts
- Phone
- Browser

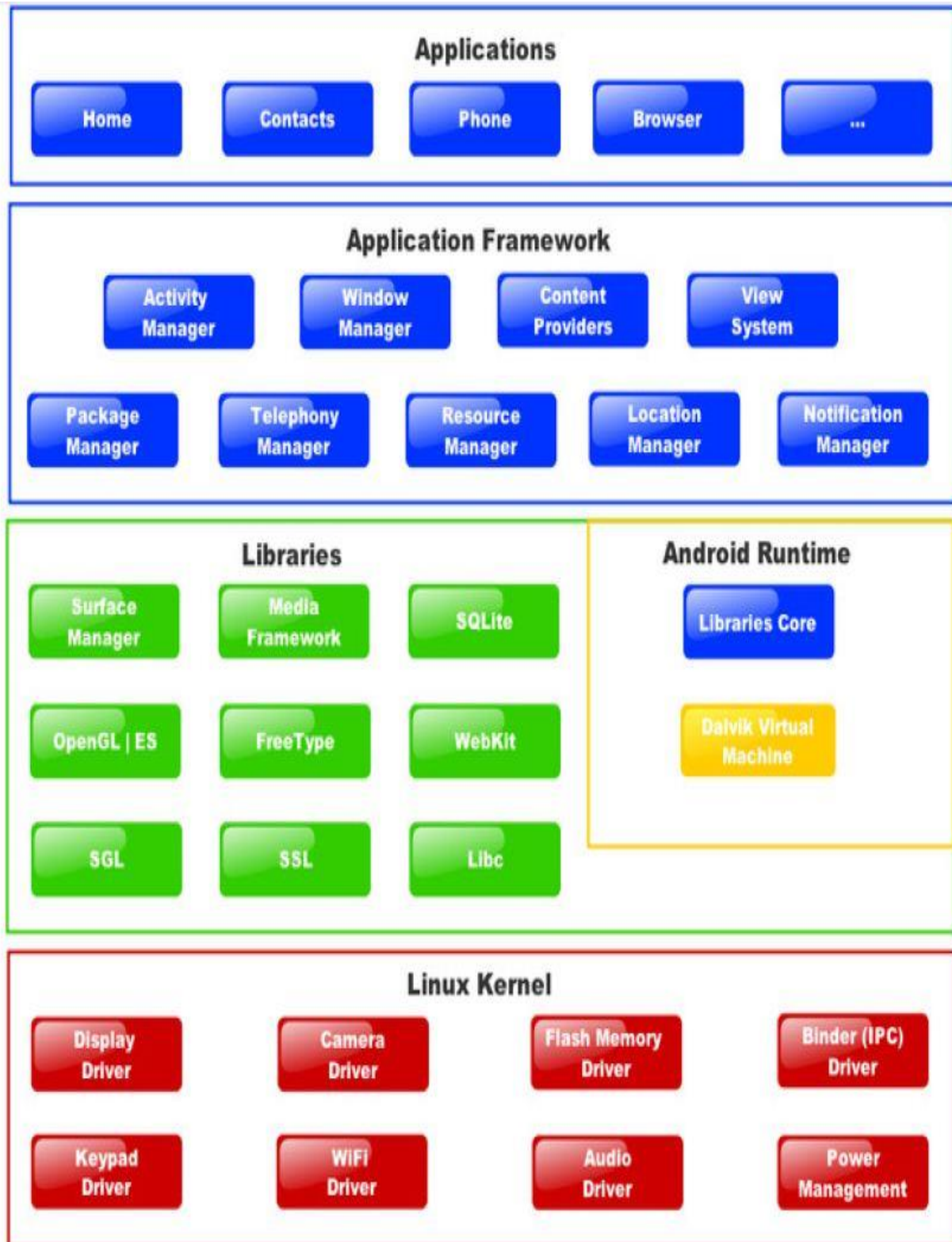


Figure 4 show Android components

3.3 Android Applications

These are the basics of Android applications:

- Android applications are composed of one or more application components (activities, services, content providers, and broadcast receivers).
- Each component performs a different role in the overall application behavior, and each one can be activated individually (even by other applications).
- The manifest file must declare all components in the application and should also declare all application requirements, such as the minimum version of Android required and any hardware configurations required.
- Non-code application resources (images, strings, layout files, etc.) should include alternatives for different device configurations (such as different strings for different languages)

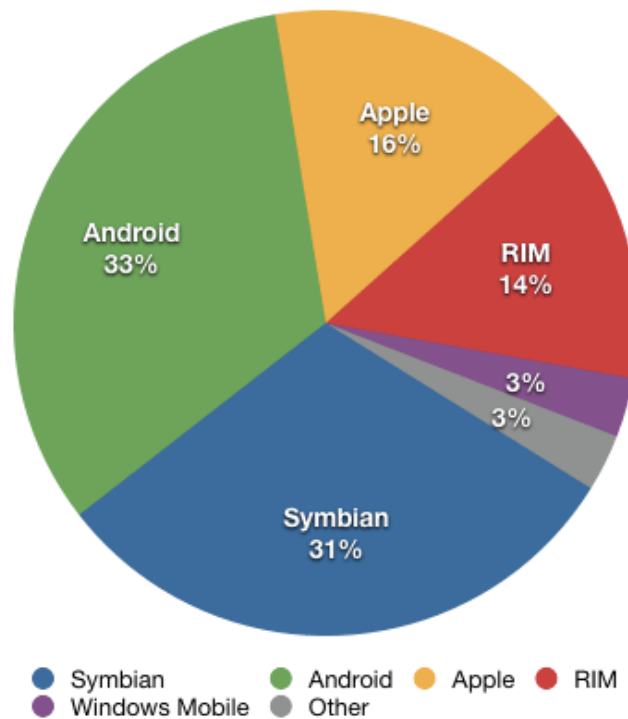
3.4 Other Mobile Operating Systems

There are different other Mobile operating systems also present in market in competition with Android. Apple's iOS and Windows Phone give strong competition to Android. A simple comparison between features and specifications of latest version of Android and other Operating Systems can be seen in the table given below.



Figure 5 show latest version of Android and other Operating Systems

According to Canalys, In Q2 2009 Android had 2.8% market share which had grown to 33% market share by Q4 2010 which made Android leader of smart phone OSs worldwide. The market share for commonly used mobile OSs is shown in the following pie chart.



A large no of mobile companies are using Android. A list of companies supporting Android in their hardware is:

- Acer Inc
- ALCATEL (TCL corporation)
- Bluelans Communications
- NCE casio Mobile Communications
- Cherry Mobile
- CSL
- Dell
- Garmin
- Geeks Phone
- General Mobile
- High screen
- HKC
- HTC coroporation
- Huawei

- **I-mobile**
- **Lenovo**
- **LG**
- **Motorola**
- **Samsung**
- **Sony Ericsson**
- **Videocon**
- **ZTE**

3.5 Android's Latest

Android is still updating. The recent version of Android (Honey comb 3.1) has very advanced features and updated applications which are optimized for use on larger screen devices. These applications are mentioned below:

1. **Browser:** Android's new version includes a number of new features for simple, fast and convenient browsing with the Quick UI controls. It also supports popular web standards like CSS 3D, Animations and CSS fixed positioning to all sites, mobile or desktop. It also supports playback of HTML5 video content. To make it better and convenient to manage favorite content, users can now save a web page locally for offline viewing, including all styling and images.



Figure 6 show a clip of new browser

2. **Gallery:** Now the new gallery supports PTP (Picture Transfer Protocol) so that users can directly connect their cameras with Android device and transfer pictures with a single touch.
3. **Calendar:** Calendar grids are larger, for better readability and more accurate touch-targeting. Additionally, users can create a larger viewing area for grids by hiding the calendar list controls. Controls in the date picker are redesigned, making them easier to see and use.
4. **Contacts:** The Contacts app now lets you locate contacts more easily using full text search. Search returns matching results from all fields that are stored for a contact.
5. **Email:** When replying or forwarding an HTML message, The Email app now sends both plain text and HTML bodies as a multi-part mime message. This ensures that the message will be formatted properly for all recipients. Folder prefixes for IMAP accounts are now

easier to define and manage. To conserve battery power and minimize cell data usage, the application now perfectness email from the server only when the device is connected to a Wi-Fi access point.

An updated Home screen widget give users quick access to more email. Users can touch Email icon at the top of the widget to cycle through labels such as Inbox, Unread, and Starred. The widget itself is now resizable, both horizontally and vertically.

6. **Enterprise support:** Users can now configure an HTTP proxy for each connected Wi-Fi access point. This lets administrators work with users to set a proxy hostname, port, and any bypass sub domains. This proxy configuration is automatically used by the Browser when the Wi-Fi access point is connected, and may optionally be used by other apps. The proxy and IP configuration is now backed up and restored across system updates and resets.

For developers, Android's new version has extended possibilities with new capabilities that developers can build on to create new and powerful applications for tablets and smart phones. Some of the new features for developers are:

- Open Accessory API for rich interaction with peripherals
- USB Host API
- Input from mice , Joystick and game pad
- Resizable home screen widgets
- MTP(Media transfer protocol) ATI for integrating with external cameras
- Real Time Transport – protocol (RTP) ATI for control over audio streaming sessions.

With all these new features Android 3.1 is making life more comfortable and advanced. Now with advanced browser, gallery, calendar, USB support, new hardware's supporting this OS are pretty convenient.

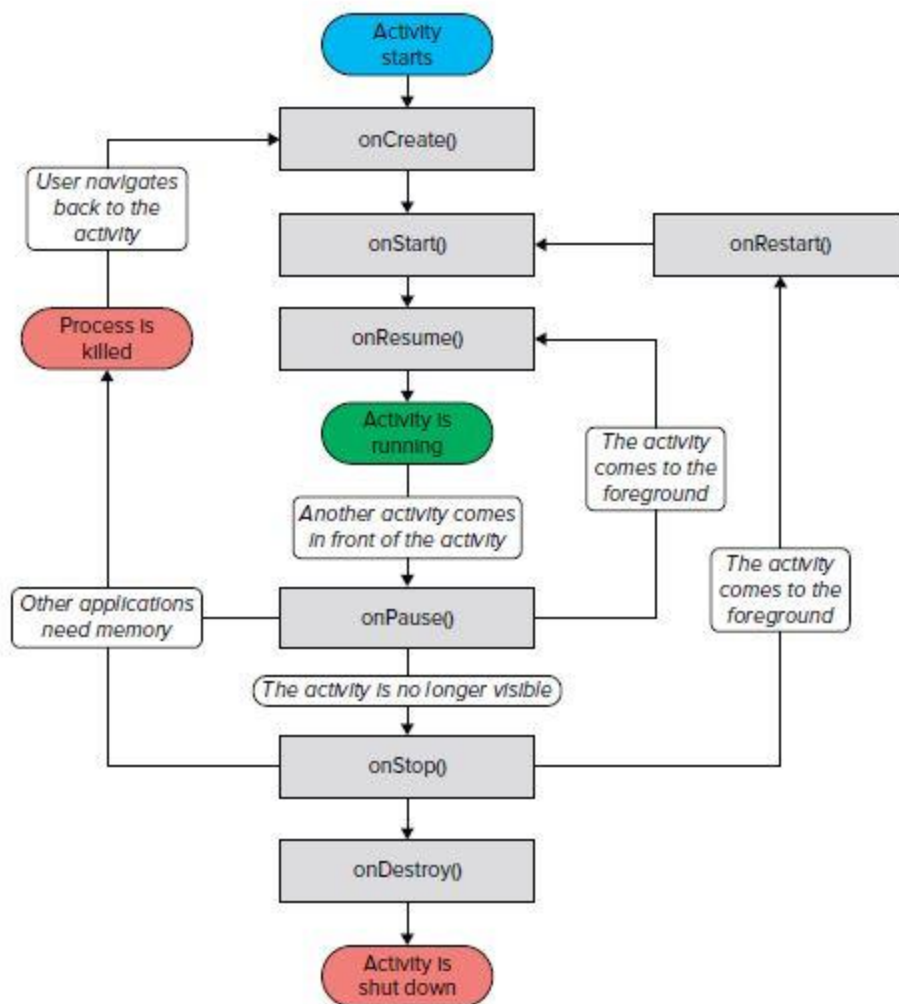
Cell phones have really evolved earlier features like camera and music player but now latest hardware's not only support such features but advanced use of such features as well as Bar code scanner, text scanner etc. which has made life better and simpler and Android has played an instrumental role in this evolution with thousands of applications available in Android market and open chance for developers to make it more advanced and better. In a way Android has changed its users' life completely.

Chapter Four

4.1 The life cycles of an activity

The learned that an activity is a window that contains the user interface of your application. An application can have zero or more activities. Typically, applications have one or more activities; and the main purpose of an activity is to interact with the user. From the moment an activity appears on the screen to the moment it is hidden, it goes through a number of stages, known as an activity's *life cycle*. Understanding the life cycle of an activity is vital to ensuring that your application works correctly.

Apart from activities, another unique concept in Android is that of an *intent*. An intent is basically the “glue” that enables different activities from different applications to work together seamlessly, ensuring that tasks can be performed as though they all belong to one single application.



Security

Linux is a highly secure system, having been tried and tested through some very harsh environments over the decades. Android heavily relies on Linux for security. All Android applications run as separate Linux processes with permissions set by the Linux system. As such, Android passes many security concerns to the underlying Linux system.

Features

Linux comes with a lot of very useful features. Android leverages many of them, such as support for memory management, power management, and networking.

Native Libraries

The native libraries are C/C++ libraries, often taken from the open source community in order to provide necessary services to the Android application layer. Among others, they include:

- ❖ *Webkit* A fast web-rendering engine used by Safari, Chrome, and other browsers
- ❖ *SQLite* A full-featured SQL database
- ❖ *Apache Harmony* An open source implementation of Java
- ❖ *OpenGL* 3D graphics libraries
- ❖ *OpenSSL* The secure locket layer

Although many of these libraries are used as-is, one notable exception is Bionic, which is basically a rewritten version of the standard C library. Bionic is used for two reasons:

Technology

To make it purpose-built for tiny, battery-powered devices

License

To make it license-friendly for others who might want to adopt it and change it

Dalvik

Dalvik is a purpose-built virtual machine designed specifically for Android, developed by Dan Bornstein and his team at Google. The Java virtual machine (VM) was designed to be a one-size-fits-all solution, and the Dalvik team felt they could do a better job by focusing strictly on mobile devices. They looked at which constraints specific to a mobile environment are least likely to change in the near future. One of these is battery life, and the other is processing power. Dalvik was built from the ground up to address those constraints.

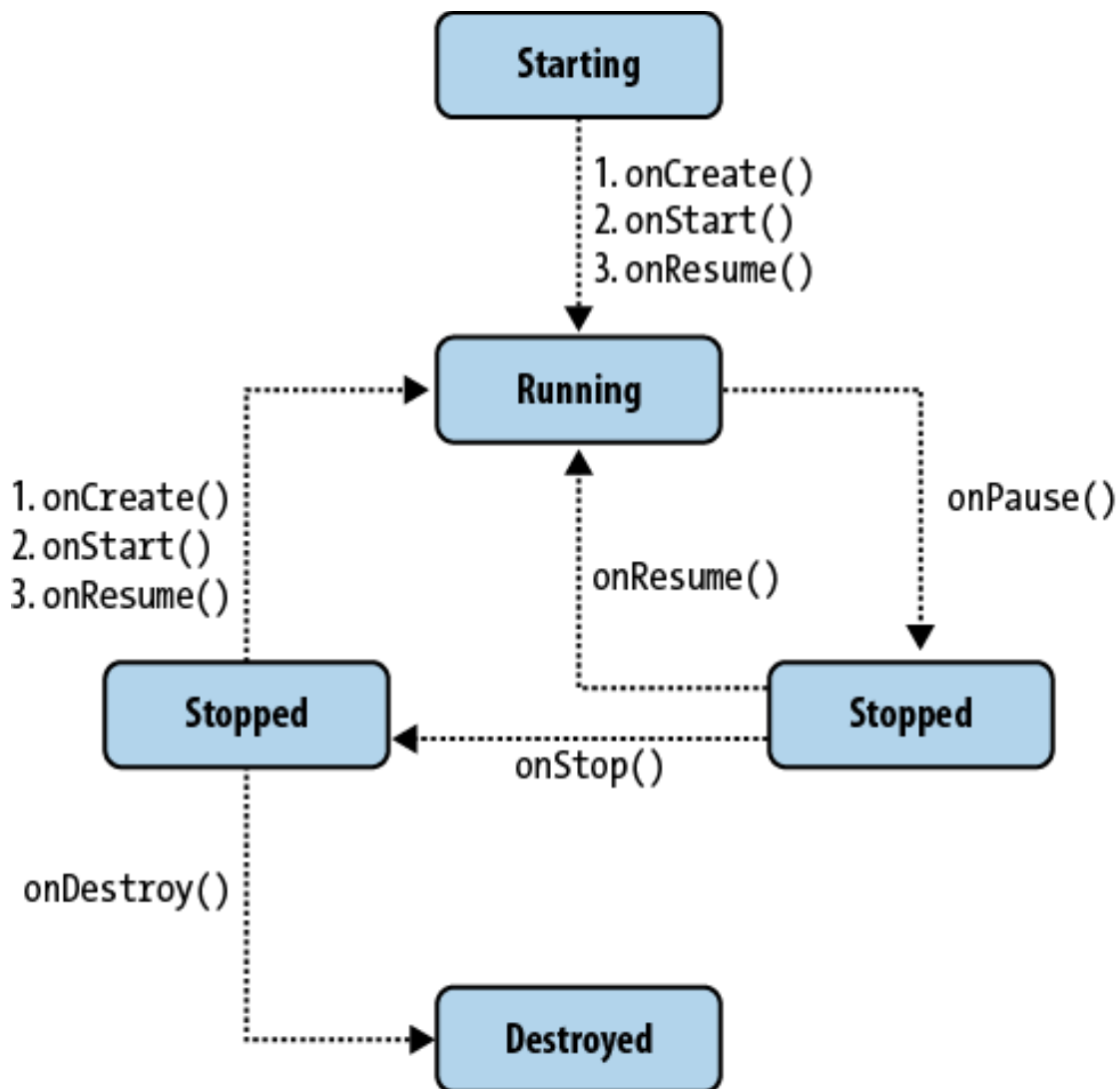
Activities

An activity is usually a single screen that the user sees on the device at one time. An application typically has multiple activities, and the user flips back and forth among them. As such, activities are the most visible part of your application. I usually use a website as an analogy for activities. Just like a website consists of multiple pages, so does an Android application consist of multiple activities? Just like a website has a “home page,” an Android app has a “main” activity, usually the one that is shown first when you launch the application. And just like a website has to provide some sort of navigation among various pages, an Android app should do the same.

Activity Life Cycle

Launching an activity can be quite expensive. It may involve creating a new Linux process, allocating memory for all the UI objects, inflating all the objects from XML layouts, and setting up the whole screen. Since we’re doing a lot of work to launch an activity, it would be a waste to just toss it out once the user leaves that screen. To avoid this waste, the activity life cycle is managed via Activity Manager. Activity Manager is responsible for creating, destroying, and managing activities. For example, when the user starts an application for the first time, the Activity Manager will create its activity and put it onto the screen. Later, when the user switches screens, the Activity Manager will move that previous activity to a holding place. This way, if the user wants to go back to an older activity, it can be started more quickly.

Older activities that the user hasn’t used in a while will be destroyed in order to free more space for the currently active one. This mechanism is designed to help improve the speed of the user interface and thus improve the overall user experience.



Intents

Intents are messages that are sent among the major building blocks. They trigger an activity to start up, tell a service to start or stop, or are simply broadcasts. Intents are asynchronous; meaning the code that sends them doesn't have to wait for them to be completed.

Intent could be explicit or implicit. In an explicit intent, the sender clearly spells out which specific component should be on the receiving end. In an implicit intent, the sender specifies the type of receiver. For example, your activity could send an intent saying it simply wants someone to open up a web page. In that case, any application that is capable of opening a web page could “compete” to complete this action.

When you have competing applications, the system will ask you which one you'd like to use to complete a given action. You can also set an app as the default. This mechanism works very similarly to your desktop environment, for example, when you downloaded Firefox or Chrome to replace your default Internet Explorer or Safari web browsers.

This type of messaging allows the user to replace any app on the system with a custom one. For example, you might want to download a different SMS application or another browser to replace your existing ones. Figure shows how intents may be used to “jump” between various activities, in the same application or in another app altogether.

