

Samsung  
TECH INSTITUTE



# Módulo 4. Introducción a Android

José A. Montenegro  
monte@lcc.uma.es



# Módulo 4. Introducción a Android

Introducción

¿Qué es Android?

Versiones de Android

El problema de la fragmentación

# **INTRODUCCIÓN**

# Dispositivos Móviles

Definición Wikipedia:

Los dispositivos móviles son aparatos:

- Pequeño tamaño
  - Capacidad de procesamiento
  - Conexión permanente o intermitente a una red
  - memoria limitada
  - Diseñados específicamente para una función,
- ¿Son estas características válidas hoy en día?



# Dispositivos Móviles

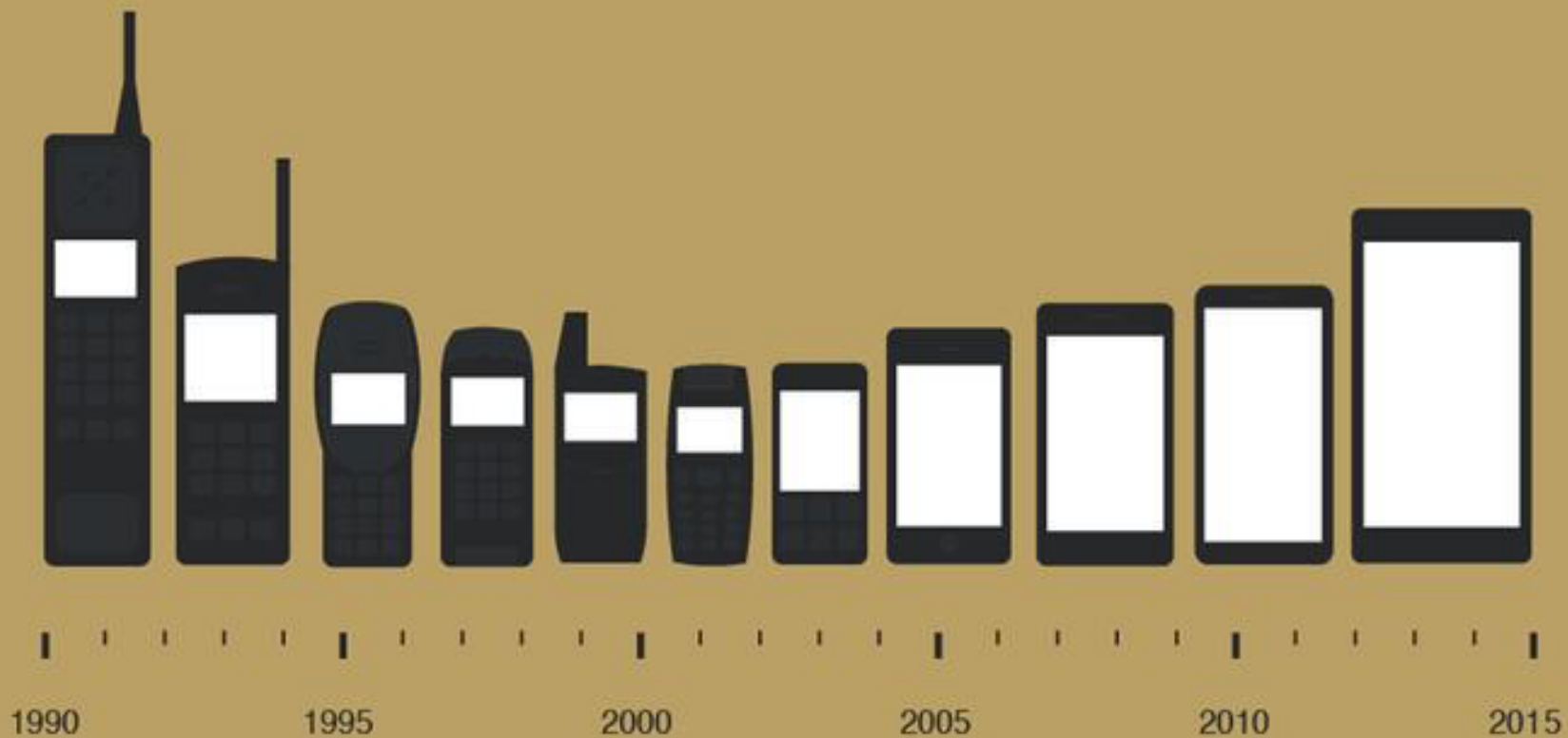
Definición Wikipedia:

**Dispositivo móvil,**

- tamaño pequeño,
- con capacidades de procesamiento,
- con [conexión a Internet](#) ,
- con [memoria](#), diseñado específicamente para una [función](#),
- pero que pueden llevar a cabo otras funciones más generales



# Evolución Terminales



# Evolución Terminales



1970  
Primer Teléfono



1992  
Nokia 101



1996  
Motorola StarTAC



1996  
Nokia Communicator



1999  
BlackBerry 850

# Evolución Terminales



2003  
Nokia 1100  
Teléfono Más Vendido



2000  
Sharp J-Phone  
Primera Cámara



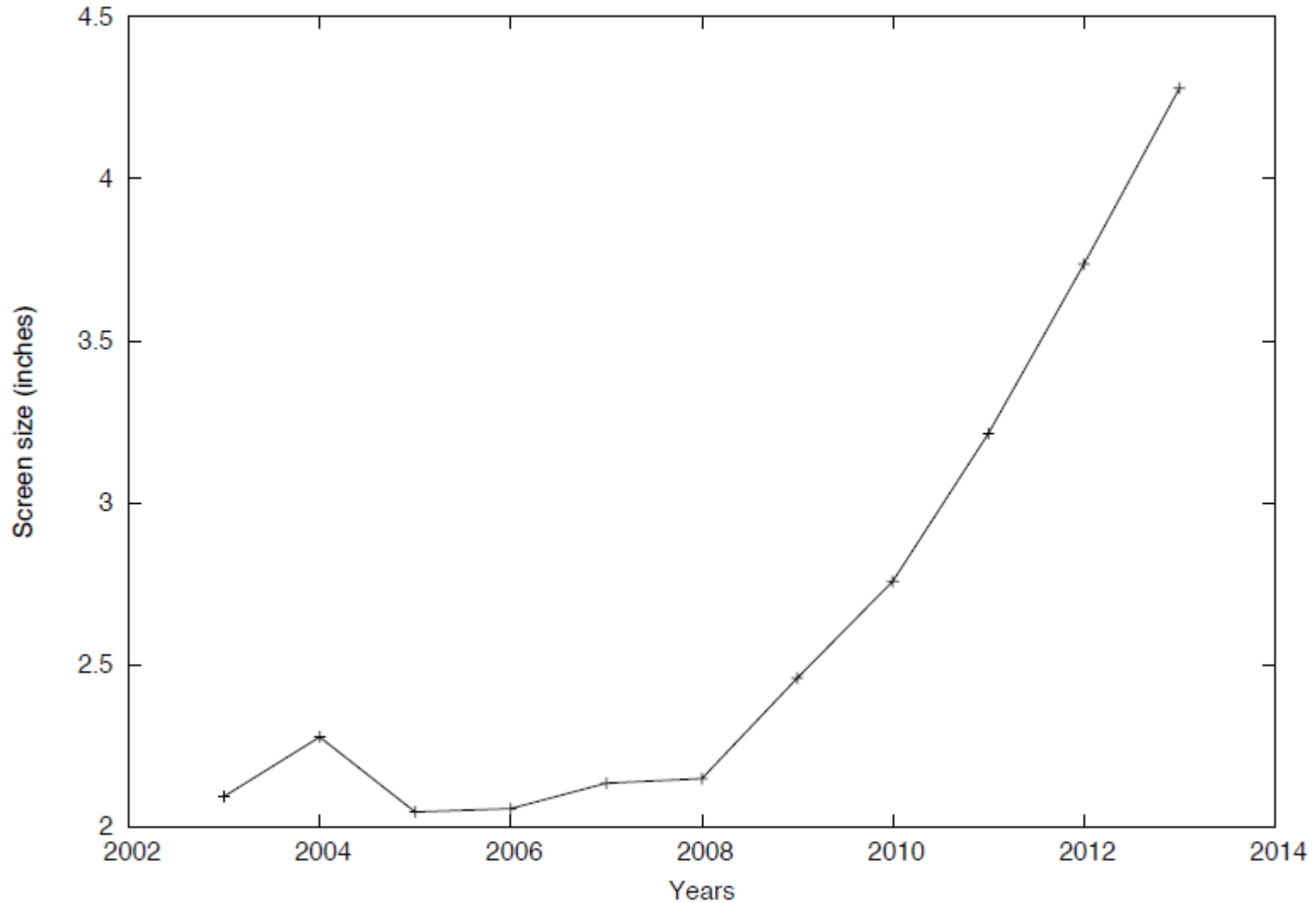
2007  
Primer  
Iphone



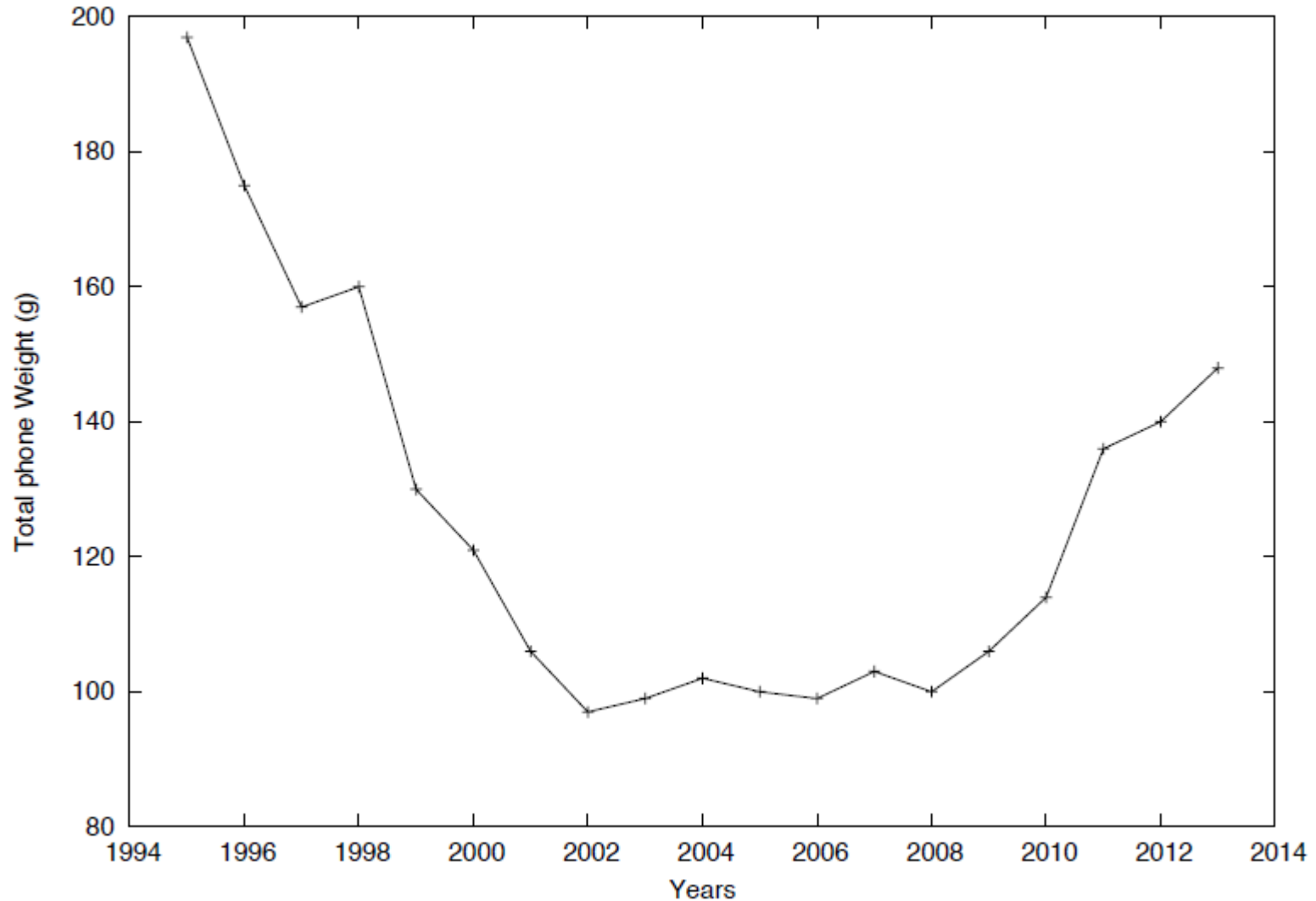
2009  
Htc dream  
Android



# Evolución Terminales (Pantalla)

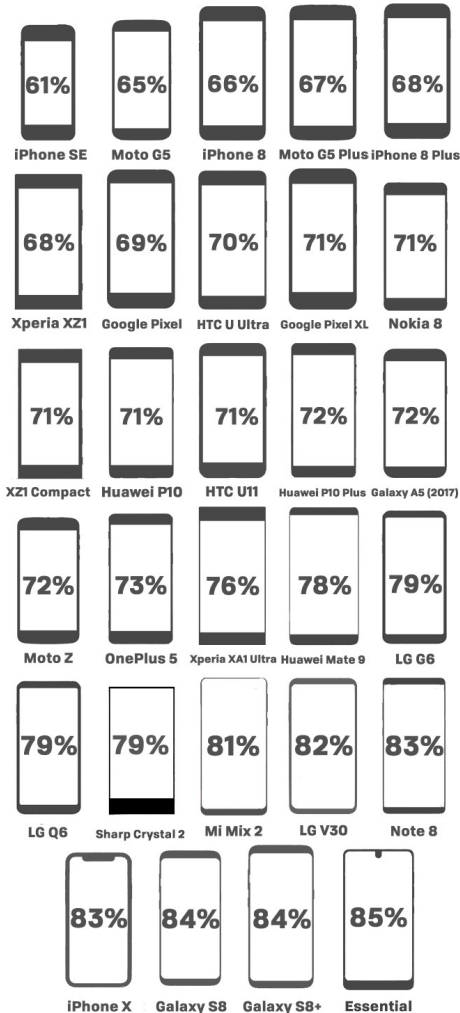
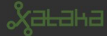


# Evolución Terminales (Peso)



# Papel de la Pantalla

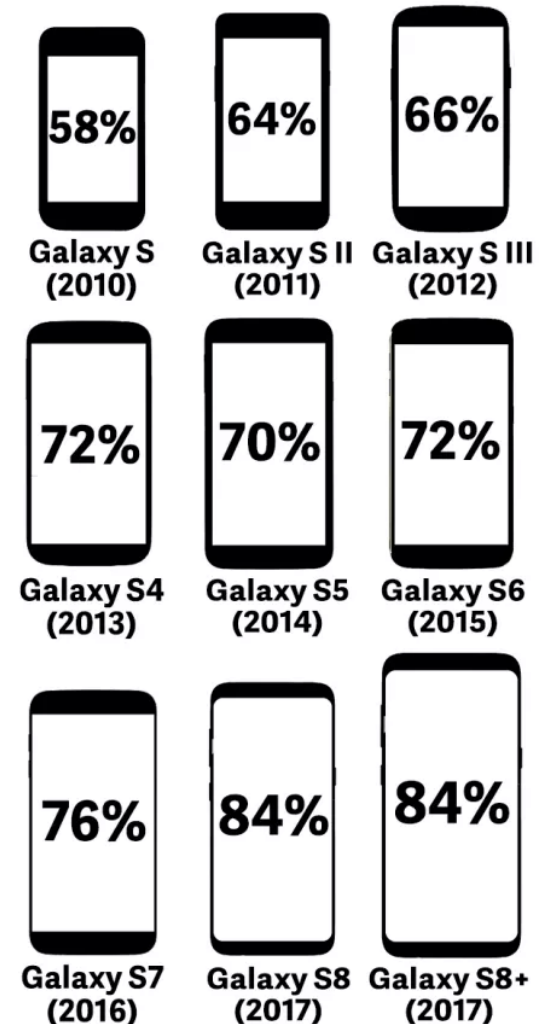
¿Qué porcentaje del frontal de tu smartphone es pantalla?



## Hacia un Galaxy S full screen



La proporción de pantalla vs marco frontal ha aumentado notablemente.



# Plataformas de desarrollo



Symbian S60



Symbian UIQ

# Plataformas de desarrollo



Windows Mobile



Windows Phone

# Plataformas de desarrollo



PalmOs v1-v3



PalmOs v3-v6

# Plataformas de desarrollo



BlackBerry v4

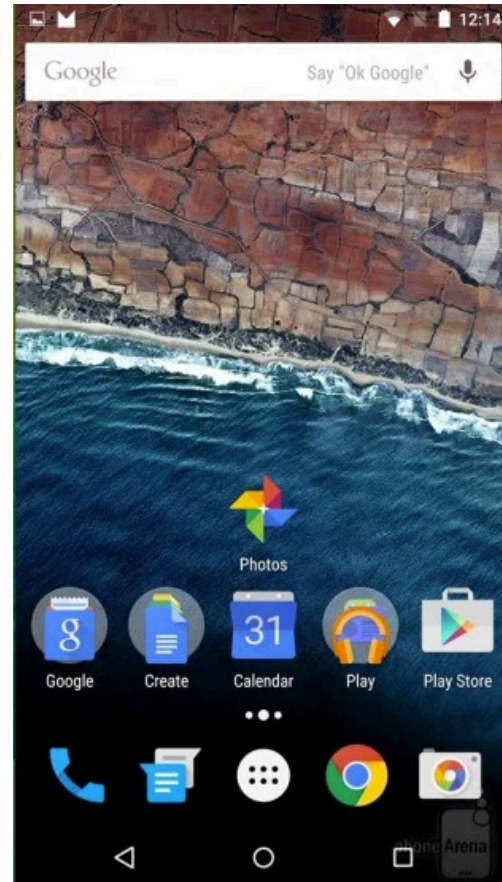


BlackBerry v 6

# Plataformas de desarrollo



iOS



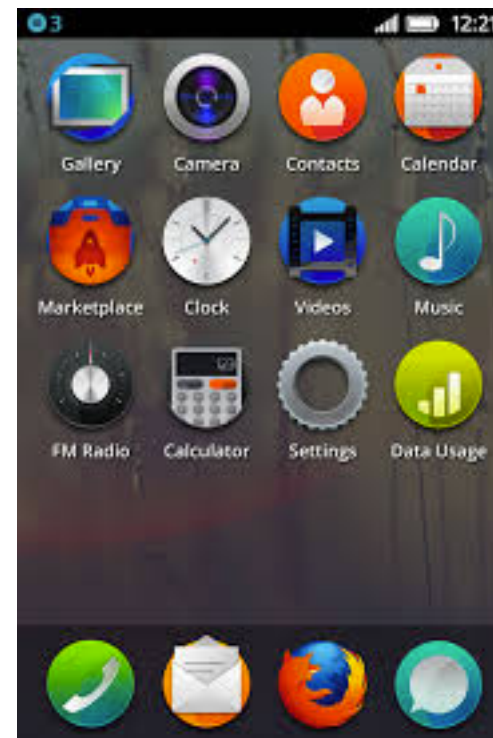
Android



# Plataformas de desarrollo



Tizen

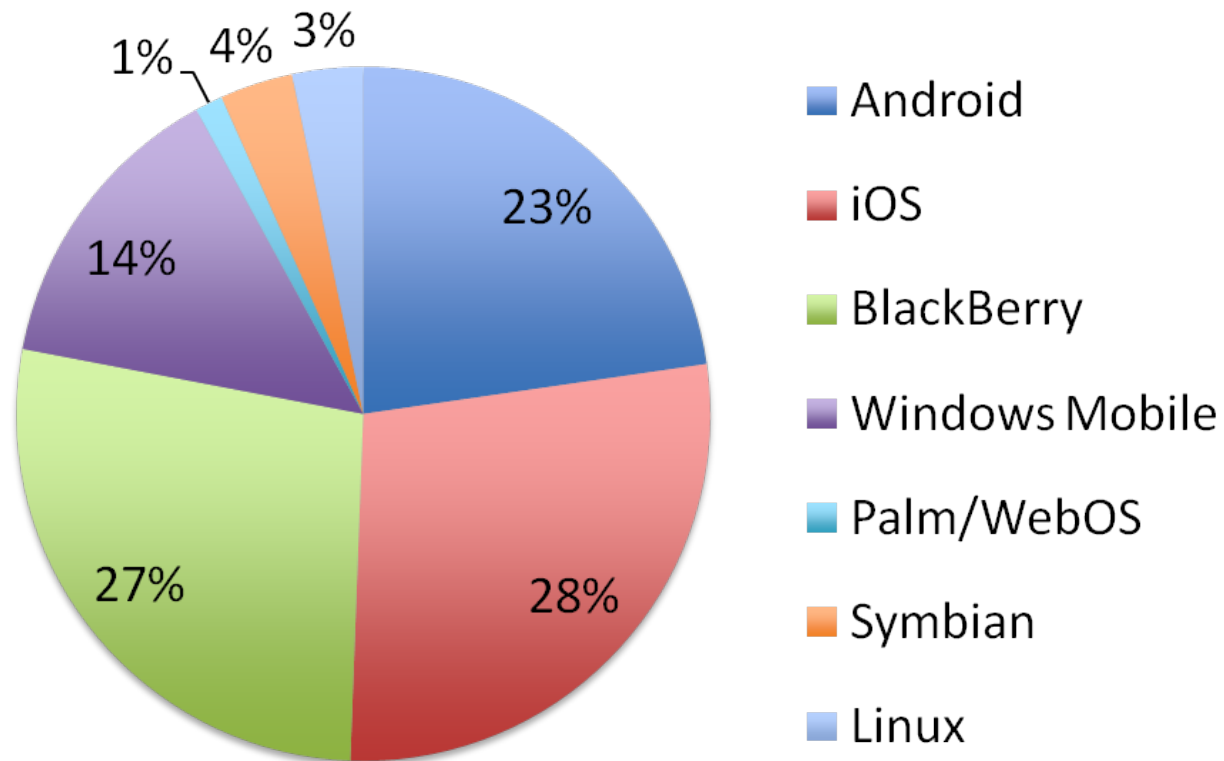


Firefox

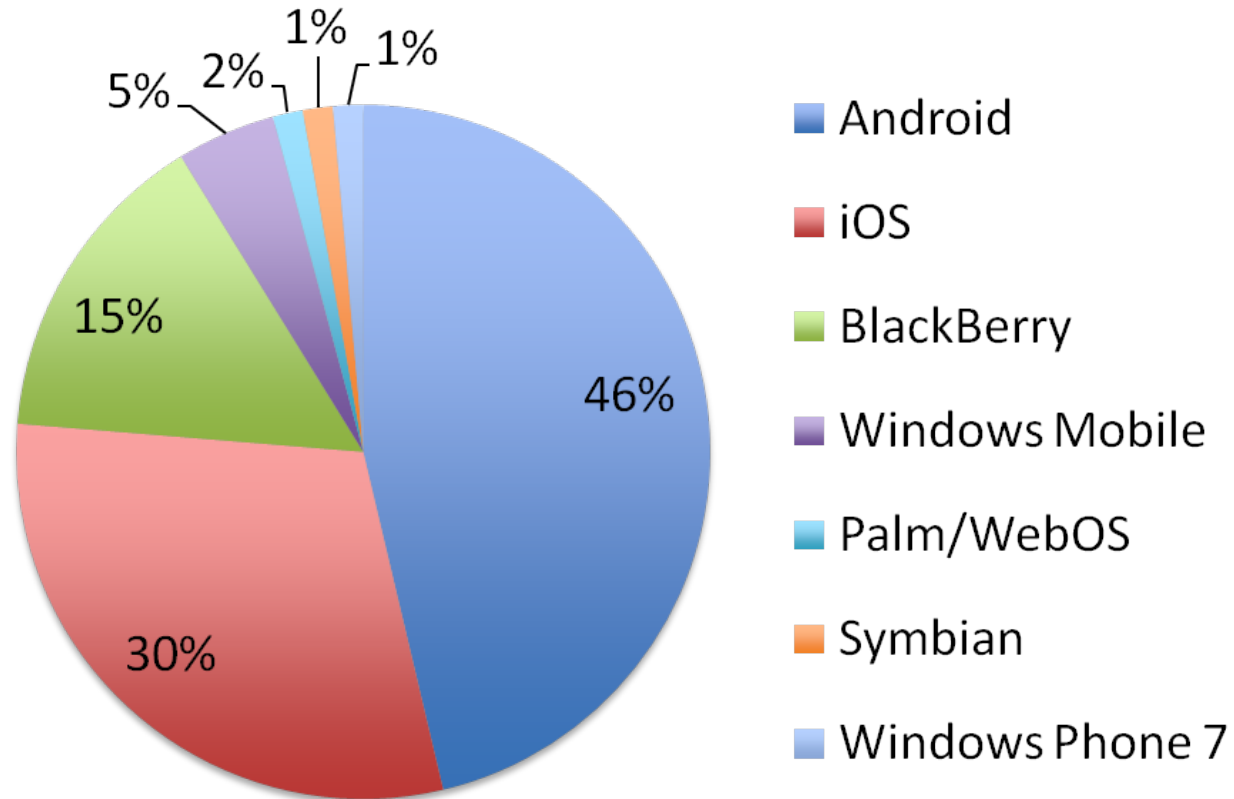
# ¿Por qué Android?

- De todas las plataformas de desarrollo existentes nos podemos preguntar la causa de elegir Android.
- Existe muchos parámetros para establecer la selección, por ejemplo desde punto vista software:
  - Código Libre.
  - Utilización sistema Linux como base.
  - Programación Java principalmente.  
Posibilidad de programación a bajo nivel si necesita más eficiencia.
  - Emuladores gratis y Documentación abundante.  
etc...
- Pero si nos fijamos en los mercados, en las siguientes transparencias podemos observar la evolución de las plataformas en el mercado Norte Americano.

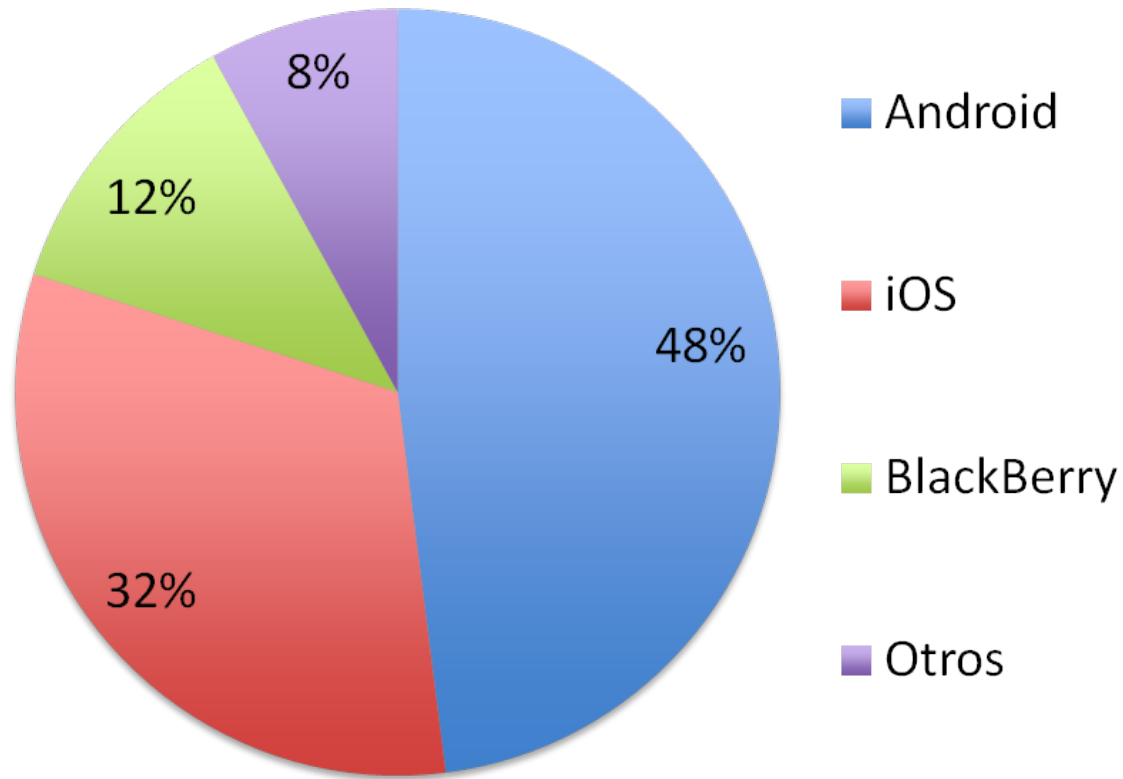
# Situación Mercado Plataformas 2010



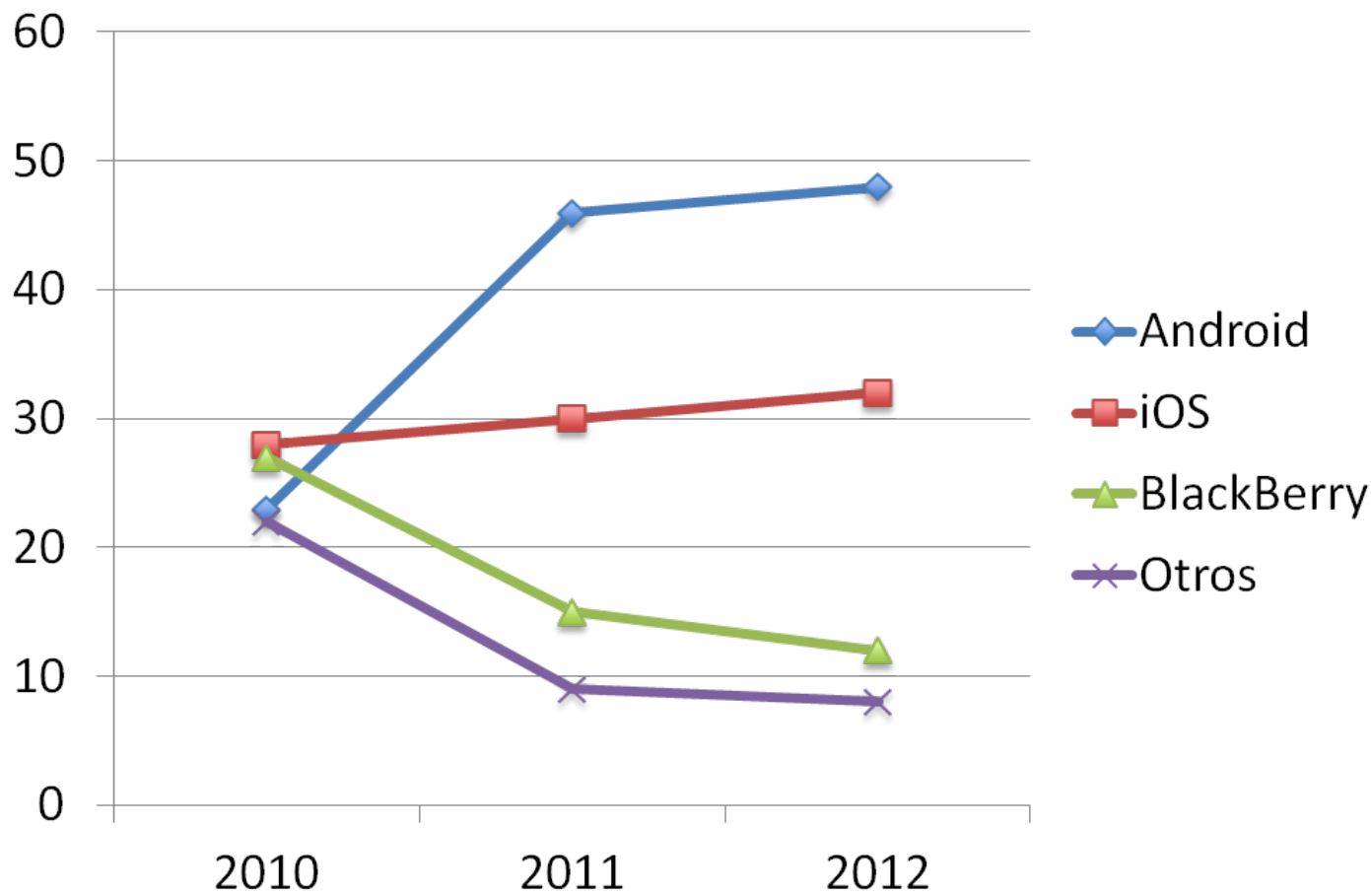
# Situación Mercado Plataformas 2011



# Situación Mercado Plataformas 2012



# Evolución Mercado 2010-2012



# Práctica 4.1

## Evolución Mercado en Smartphones

**¿QUÉ ES ANDROID?**



# ¿Qué es Android?

- Android es una "pila" software para dispositivos móviles que incluyen un sistema operativo, middleware y aplicaciones esenciales.
- El Kit de desarrollo Software (SDK) de Android proporciona las herramientas y los interfaces de programación (APIs) necesarias para comenzar a desarrollar aplicaciones en la plataforma Android con el lenguaje de programación Java.

También es posible hacer uso de otros lenguajes como C, C++ para programar una aplicación en Android.

# ¿Qué es Android?

- Marco de Trabajo de Aplicación permite reutilizar y reemplazar los componentes.
- Máquina Virtual Dalvik máquina virtual Java optimizada para dispositivos móviles.
- Navegador web integrado, basado en el proyecto opensource WebKit.
- Gráficos Optimizados mediante una librería 2D y 3D basados en la especificación OpenGL
- SQLite para el almacenamiento de datos.
- Soporte Multimedia para audio, vídeo, e imágenes
- Telefonía GSM, 3G, 4G, Bluetooth y WiFi
- Cámara, GPS, Brújula, y Acelerómetro
- Entorno de desarrollo completo incluye un emulador de dispositivo, herramientas para depurar. (Plugin para el IDE de Eclipse).
- .....

# Arquitectura



**Aplicaciones**

**Marco Trabajo Aplicaciones**

**Bibliotecas y Entorno Ejecución**

**Kernel de Linux**

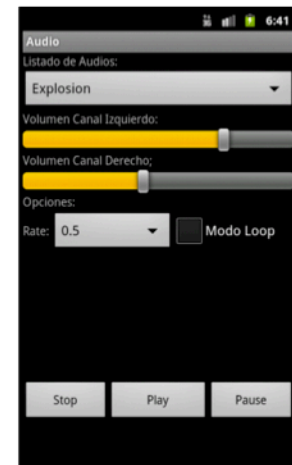
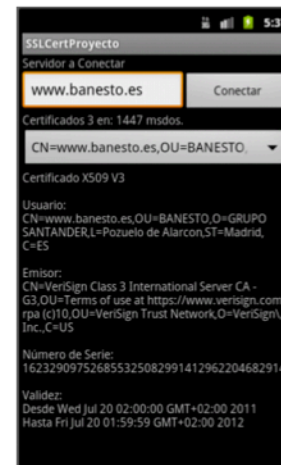
# Arquitectura

## Aplicaciones

Marco Trabajo Aplicaciones

Bibliotecas y Entorno Ejecución

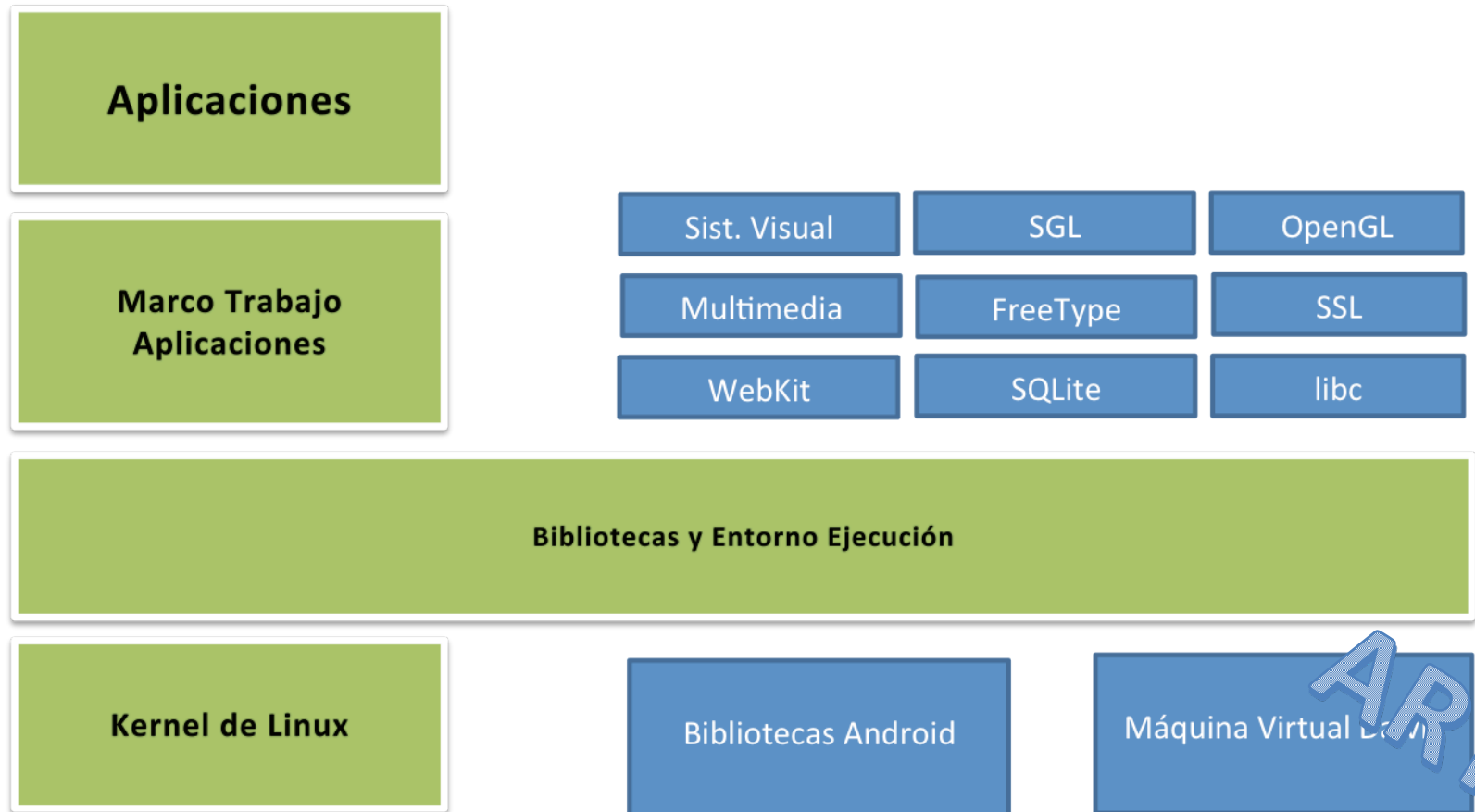
Kernel de Linux



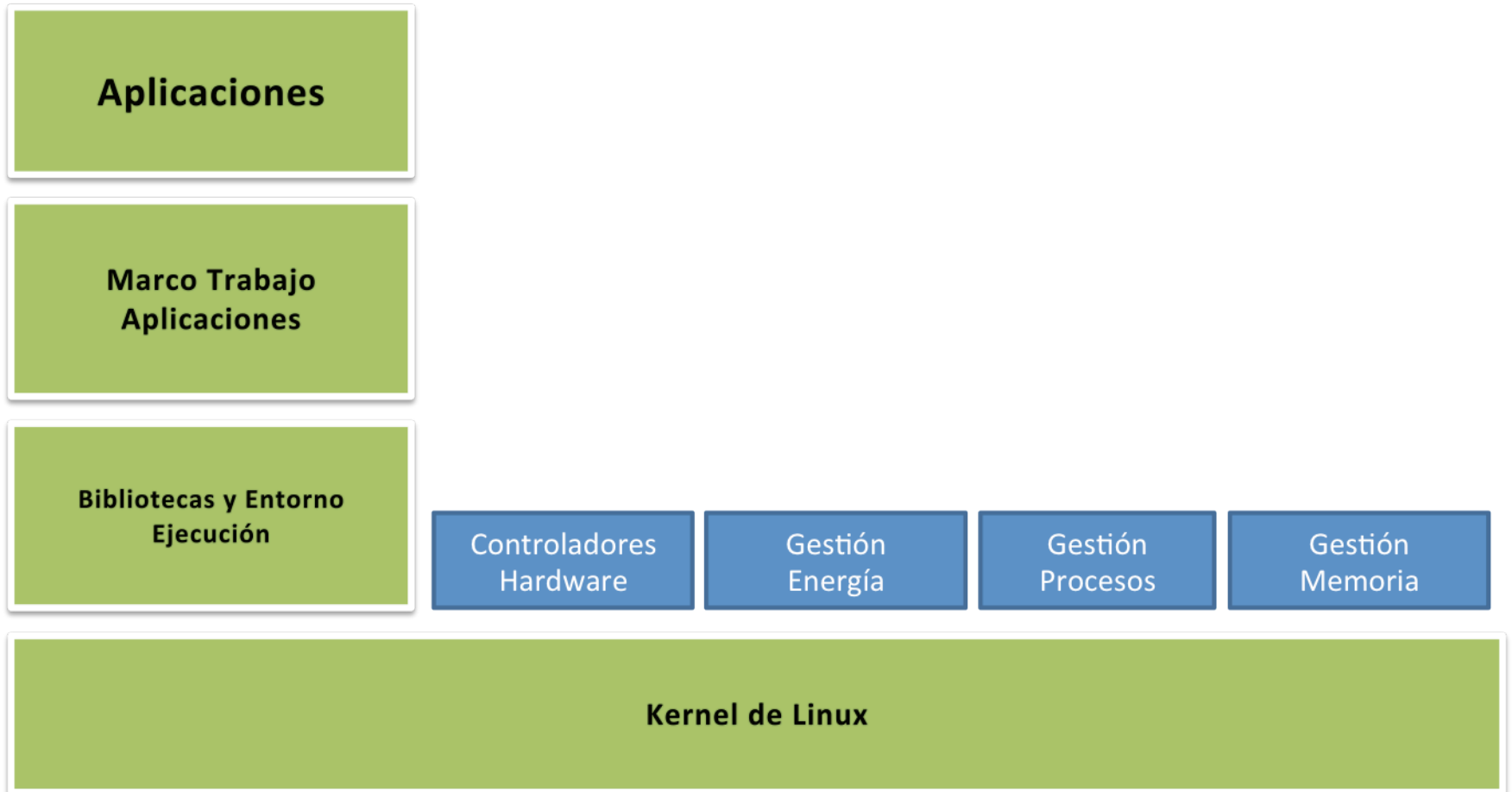
# Arquitectura



# Arquitectura



# Arquitectura



# Arquitectura



Aplicaciones

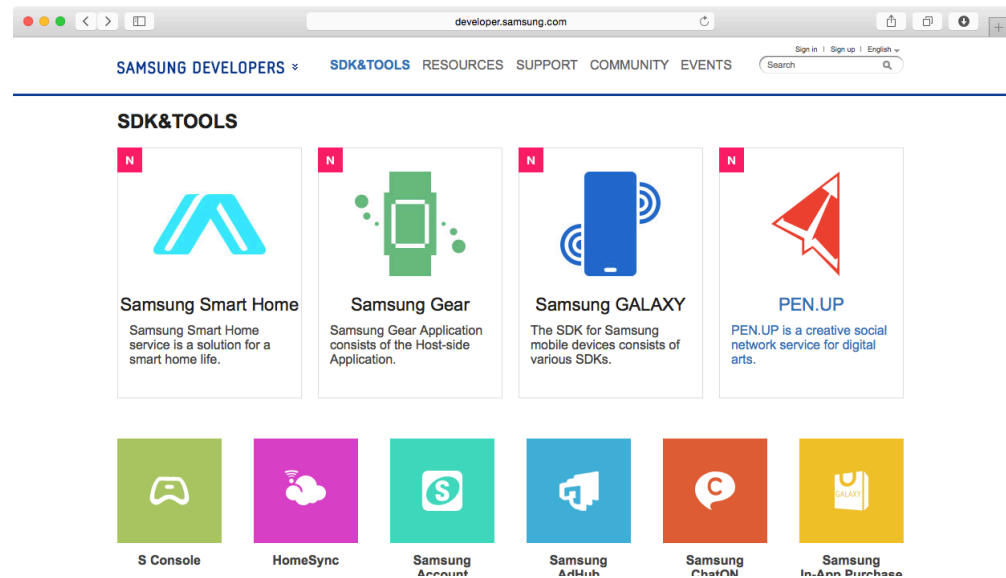
Marco Trabajo Aplicaciones

Bibliotecas y Entorno Ejecución

Kernel de Linux

## Samsung TouchWiz

<http://developer.samsung.com/sdk-and-tools>





# Y mas..



## Las API de Google para Android

Simplifica la programación de tu Android, aumenta la base de usuarios y monetiza de manera más eficaz con los servicios de Google.

INICIO

GUÍAS

REFERENCIA



### Programación

**Envía aplicaciones de mayor calidad, más rápido.**

Obtén una ventaja en la programación con los servicios de Google que puedes usar de inmediato.

[Analytics](#)

### Atrae usuarios

**Desarrolla una base de usuarios activa.**

Llega a nuevos usuarios y encuentra maneras de que regresen una y otra vez a tu aplicación.

[Envío de mensajes a través de la nube](#)

[Indexación de aplicaciones](#)

### Gana dinero

**Utiliza una variedad de herramientas para ganar más dinero.**

Utiliza un enfoque personalizado para cada usuario en relación con la monetización a fin de maximizar los ingresos de la aplicación.

[AdMob](#)

developers.google.com

# Práctica 4.2

## Interfaz Propietarias y Google APIs

# Ejecución Aplicación Java vs Android

- Normalmente las aplicaciones en Java tiene un único punto de Ejecución.
- La maquina virtual de Java ejecutará el método Main.

```
public class HolaMundo{  
    public static void main(String args[]){  
        System.out.println("Hola Mundo!");  
    }  
}
```

# Ejecución Aplicación Java vs Android

- Sin embargo en Android, las aplicaciones no tiene un único punto de ejecución.
- Dependen de las definiciones en el manifiesto.

```
<application ... >
    <activity android:name=".HolaMundo" ...>
    </activity>
</application>

public class HolaMundo extends Activity{
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

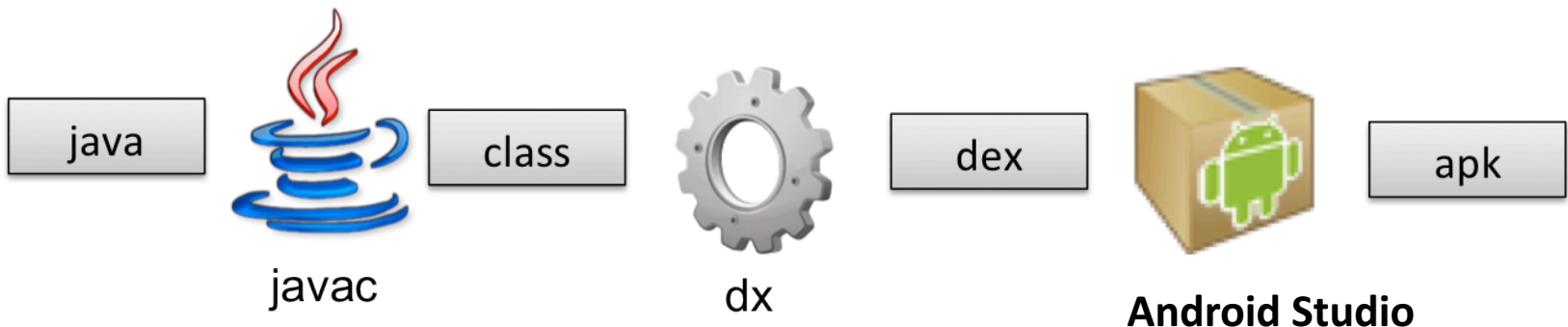
# Ejecución Aplicaciones

- **Maquina Virtual Dalvik:**

- Incluida en Android desde las primeras versiones, hasta Android 5.0.
- No es exactamente una Máquina Virtual Java, no ejecuta Java bytecode.

- **Android Runtime**

- ART reemplaza a Dalvik en Android 5.0, aunque estaba disponible desde Android 4.4.
- Transforma el bytecode en instrucciones nativas que son ejecutadas por el entorno de ejecución.



# **VERSIONES DE ANDROID**

# Versiones de Android



**Cupcake**  
Android 1.5



**Donut**  
Android 1.6



**Eclair**  
Android 2.0/2.1



**Froyo**  
Android 2.2



**KitKat**  
Android 4.4



**Lollipop**  
Android 5.0



**Gingerbread**  
Android 2.3



**Honeycomb**  
Android 3



**Ice Cream Sandwich**  
Android 4



**Jelly Bean**  
Android 4.1

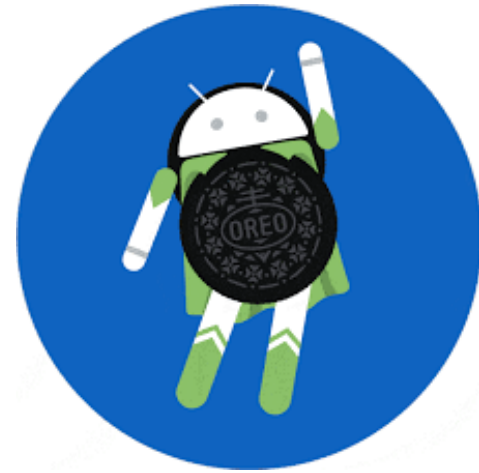
# Versiones de Android



**M -Marshmallow**  
Android 6.0



**N- Nougat**  
Android 7.0



**Open Wonder (Oreo)**  
Android 8.0



# Versiones de Android

Además nos informa del nivel de la API correspondiente a cada versión, necesaria a la hora de realizar una aplicación y mantener la compatibilidad.

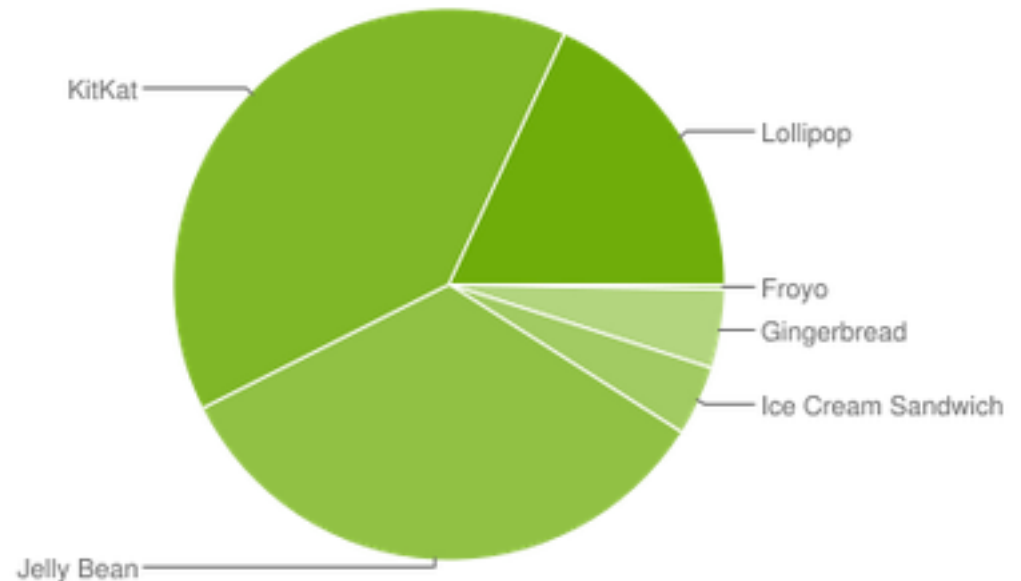
Plataforma	Nombre	Nivel API
Android 1.0, 1.1	Base	1,2
Android 1.5	Cupcake	3
Android 1.6	Donut	4
Android 2.0,2.0.1,2.1	Eclair	5,6,7
Android 2.2	Froyo	8
Android 2.3 - 2.3.2, 2.3.3 - 2.3.7	Gingerbread	9,10
Android 3.0,3.1,3.2	Honeycomb	11,12,13
Android 4.0-4.0.2	Ice Cream Sandwich	14
Android 4.1-4.1.1,4.2-4.2.2,4.3	Jelly Bean	16,17,18
Android 4.4	Kitkat	19,20
Android 5, 5.1	Lollipop	20,21,22
Android 6	M	23

<http://developer.android.com/guide/topics/manifest/uses-sdk-element.html>

# Versiones de Android

3 Agosto, 2015

Version	Codename	API	Distribution
2.2	Froyo	8	0.3%
2.3.3 - 2.3.7	Gingerbread	10	4.6%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	4.1%
4.1.x	Jelly Bean	16	13.0%
4.2.x		17	15.9%
4.3		18	4.7%
4.4		19	39.3%
5.0	Lollipop	21	15.5%
5.1		22	2.6%

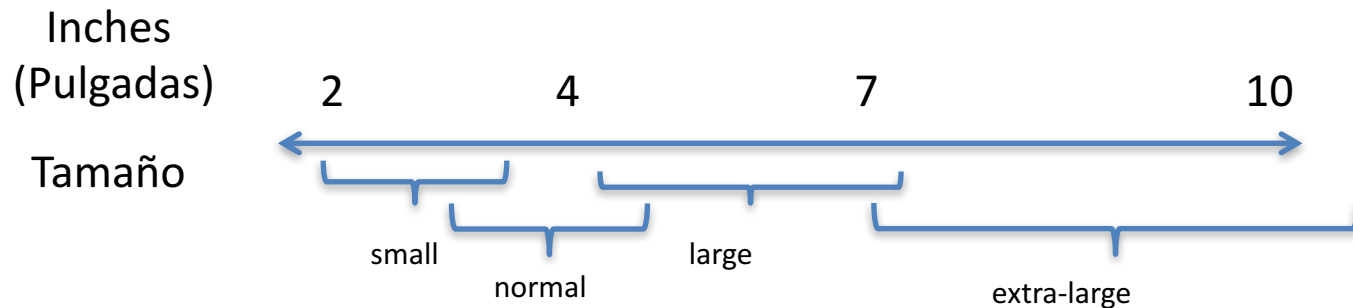


<http://developer.android.com/about/dashboards/index.html>

**MÚLTIPLES PANTALLAS**

# Soporte para Múltiples Pantallas

- **Tamaño de Pantalla:** *Tamaño físico de la pantalla, medido con la diagonal de la pantalla (inches, pulgadas)*
  - Cuatro tamaños: small, normal, large, and extra-large.



<b>xlarge</b> al menos	720dp x 960dp	<b>large</b> al menos	480dp x 640dp
<b>normal</b> al menos	320dp x 470dp	<b>small</b> al menos	320dp x 426dp

# Soporte para Múltiples Pantallas

- ***Density-independent pixel (dp).*** *Unidad virtual que es utilizada cuando desarrollamos el interfaz visual.*
  - 1 dp es equivalente a un pixel físico en una pantalla de 160 dpi.
  - La conversión es realizada en tiempo de ejecución  
 $px = dp * (dpi / 160)$ .
    - Pantalla de 240 dpi, 1dp es 1.5 pixels físicos (240/160).
  - Los elementos visuales debe ir definidos en dp para que sean mostrados correctamente en pantallas con diferentes densidades.

# Soporte para Múltiples Pantallas

- ***Densidad Pantalla (dpi):*** *La cantidad de pixels en un área de la pantalla.*
  - dpi (dots per inch- puntos por pulgada).
  - Seis grupos: low, medium, high, extra-high, extra-extra-high and extra-extra-extra-high.

*ldpi* (low) ~120dpi

*mdpi* (medium) ~160dpi

*tvdpi* ~213 dpi

*hdpi* (high) ~240dpi

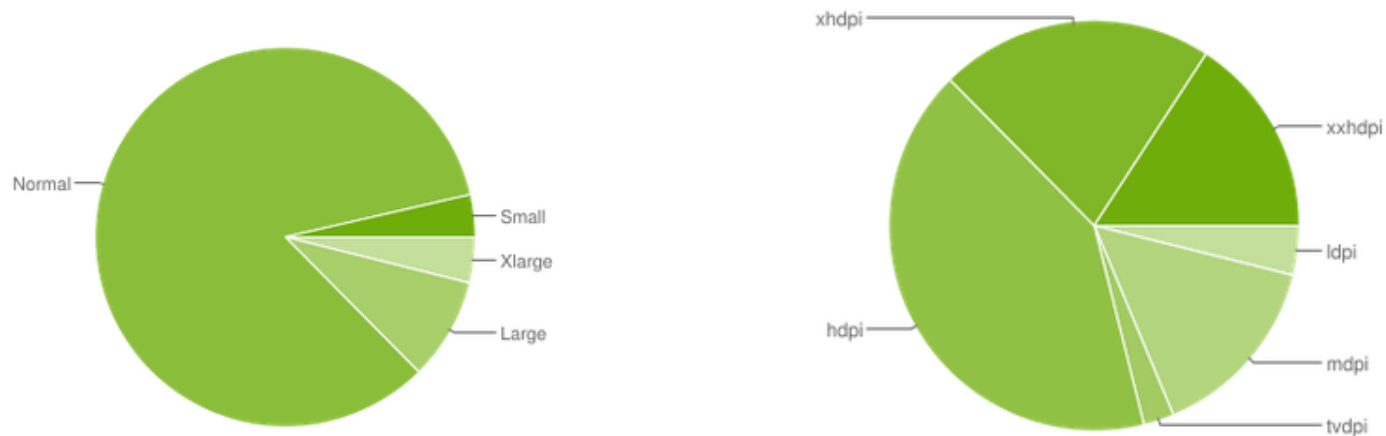
*xhdpi* (extra-high) ~320dpi

*xxhdpi* (extra-extra-high) ~480dpi

*xxxhdpi* (extra-extra-extra-high) ~640dpi

# Tamaño Pantalla y Densidad

7 Agosto, 2015



	ldpi	mdpi	tvdpi	hdpi	xhdpi	xxhdpi	Total
Small	3.6%						3.6%
Normal		6.9%	0.1%	40.6%	20.4%	15.8%	83.8%
Large	0.3%	4.9%	2.3%	0.6%	0.6%		8.7%
Xlarge		3.0%		0.3%	0.6%		3.9%
Total	3.9%	14.8%	2.4%	41.5%	21.6%	15.8%	

<http://developer.android.com/about/dashboards/index.html>

# Práctica 4.3

## Evolución Versiones Android y Tamaño de Pantallas



# **EL PROBLEMA DE LA FRAGMENTACIÓN**

# El problema de la fragmentación

- Características Específicas Operadores
- UI Específico
- Impacto a los desarrolladores
- Riesgos de Seguridad

Samsung  
TECH INSTITUTE



# Módulo 4. Introducción a Android

José A. Montenegro  
monte@lcc.uma.es

