

# Taller de Sistemas de Información 2

---



Laboratorio de Integración de Sistemas

Android  
Clase 6



# Contenidos

---

- Introducción
- Historia y actualidad
- Arquitectura
- Desarrollo
- Modelo de programación
- Interfaz de usuario
- Testing
- Referencias y recomendaciones



# Introducción

## ¿Qué es Android?

- ❑ SO basado en el kernel de Linux (2.6) GPLv2
  - ❑ 12 millones de líneas de código
    - ❑ 3M - XML
    - ❑ 2,8M - C
    - ❑ 2,1M - Java
    - ❑ 1,75M - C++.
  - ❑ Diseñado por Google para disp. móviles
  - ❑ La mayoría del código bajo licencia Apache



# Introducción

## ¿Beneficios de Android?

- Portable/adaptable a casi cualquier hardware
- Buena calidad
- Fabricantes no tienen que pagar
  - No es el único... pero...
  - Maemo
  - Ubuntu mobile



# Historia

## Hitos

- {1968} *Do Androids Dream of Electric Sheep?*
  - Philip K. Dick (1928-1982) - [ciencia ficción]
- {2005} Google compra Android fundada en 2003
  - Objetivo desarrollar plataforma móvil abierta
- {2007} Presenta a Android como SO para tels.
  - *Open Handset Alliance*



# Historia

## Hitos



# Historia

## Hitos

- {1968} *Do Androids Dream of Electric Sheep?*
  - Philip K. Dick (1928-1982) - [ciencia ficción]
- {2005} Google compra Android fundada en 2003
  - Objetivo desarrollar plataforma móvil abierta
- {2007} Presenta a Android como SO para tels.
  - *Open Handset Alliance*
- {2008} Se vende el primer móvil HTC Dream
- {2013} Objetivo de agencias de inteligencia
- {2014} Unifica hardware y software



# Historia Hoy...

android embedded

Web Imágenes Noticias Videos Más + Herramientas de búsqueda

Cerca de 41.000.000 resultados (0,20 segundos)

**Embedded Android - O'Reilly Media**  
[shop.oreilly.com/product/0636920021094.do](#) • Traducir esta página  
★★★★★ Valoración: 4,6 - 14 reseñas  
Looking to port Android to other platforms such as **embedded devices**? This hands-on book shows you how Android works and how you can adapt it to fit your ...

**[PDF] Android on a non-mobile **embedded** system - The Linux...**  
[events.linuxfoundation.org/.../Android-non-mobile....](#) • Traducir esta página  
Android on a non-mobile **embedded** system. ◦ Benefits and limitations. – Reduces time-to-market. – Is somewhat familiar. – Is sometimes supported on your ...

**Imágenes de android embedded** Notificar imágenes



[Más imágenes de android embedded](#)

**Embedded Android - Free Download eBook - pdf - IT eBooks**  
[www.it-ebooks.info/book/2073/](#) • Traducir esta página  
Looking to port Android to other platforms such as **embedded devices**? This hands-on book shows you how Android works and how you can adapt it to fit your ...

**Android OS as an **embedded** platform | ITworld**  
[www.itworld.com/.../android-os-embedded-platform](#) • Traducir esta página  
26/9/2013 - How Android is being used as an **embedded** operating system.

**Digi Embedded Application Development Kit for Android ...**  
[www.digi.com/.../ARM Development](#) • Traducir esta página  
Digi Embedded Application Development Kit for Android is an ideal software platform to create professional and feature-complete products with significantly ...



# Historia

## Creador – Andy Rubin (1962)

- Nueva York
- Horace Greeley High School, Chappaqua (1977–1981)
- Utica College, Lic. C. de Computación, Utica (1981–1986)



*"I want to find something that delights people so they use it..." November 4, 2007*

# Historia

## Creador – Andy Rubin (1962)

- Carl Zeiss AG (robótica 1986–1989)
- Bill Caswell → Apple Inc. en (fabrica 1989–1992)
- General Magic (Apple Spin-off 1992–1995)
  - Magic Cap, SO e interfaz para *hand-helds*.
- Artemis Research (1995–1999)
  - Steve Perlman ex Apple (QuickTime, ...)
  - Cambia a WebTV
  - Adquirida por MS(1997) - 425M (MSN TV)



# Historia

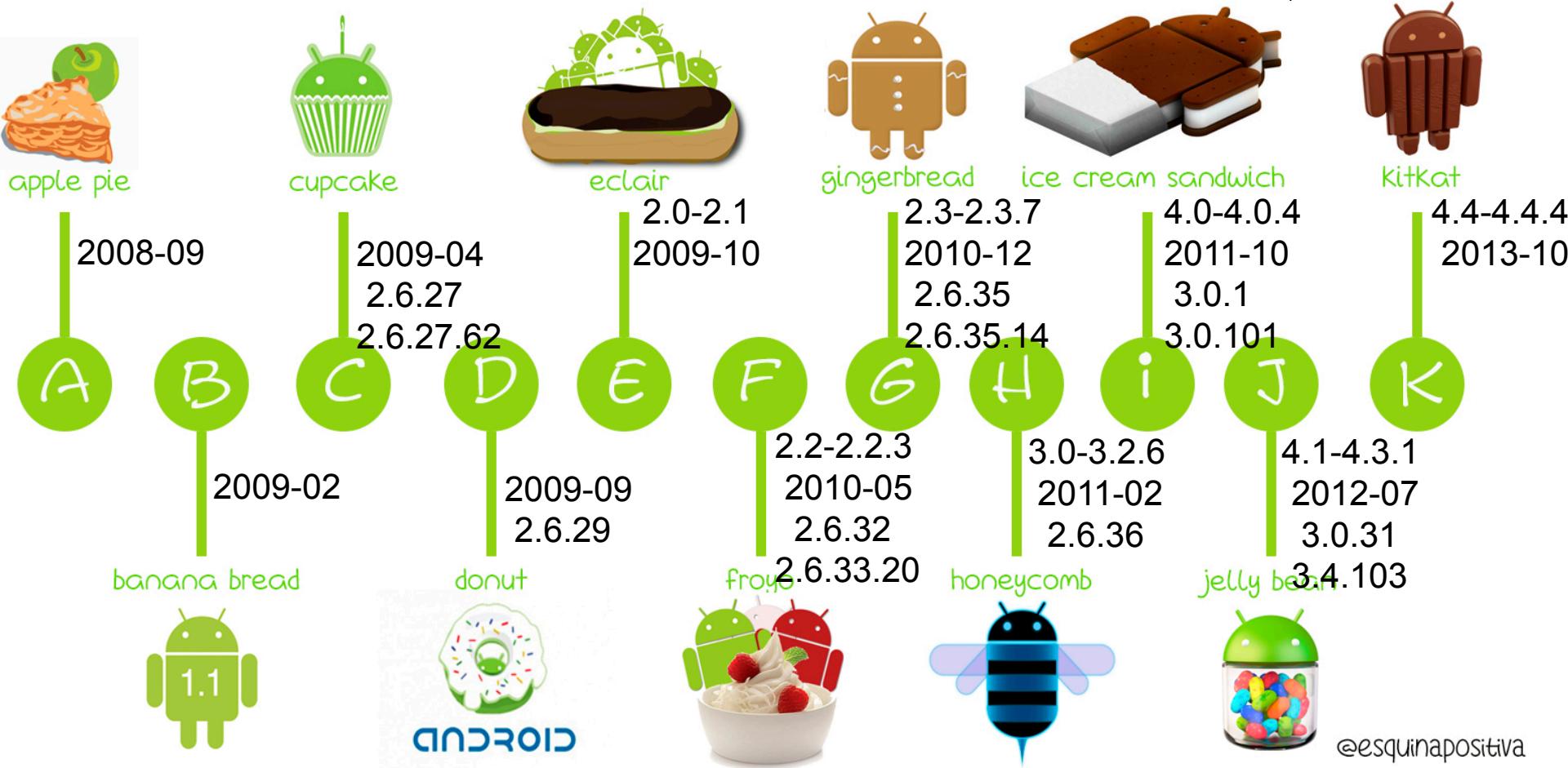
## Creador – Andy Rubin (1962)

- Danger Inc. (1999–2003)
  - Matt Hershenson+Joe Britt ex Apple hoy Google
  - Reconocida por Hiptop
    - Teléfono similares a PDA
  - Adquirida por MS (2008) - 500M
- Android Inc. (2003–2005) - ¿M?
- Vicepresidente de ingeniería en Google (2005–...)
- Más de 15 patentes (Danger, Google)
- Comunicación, conversiones, activaciones, ...

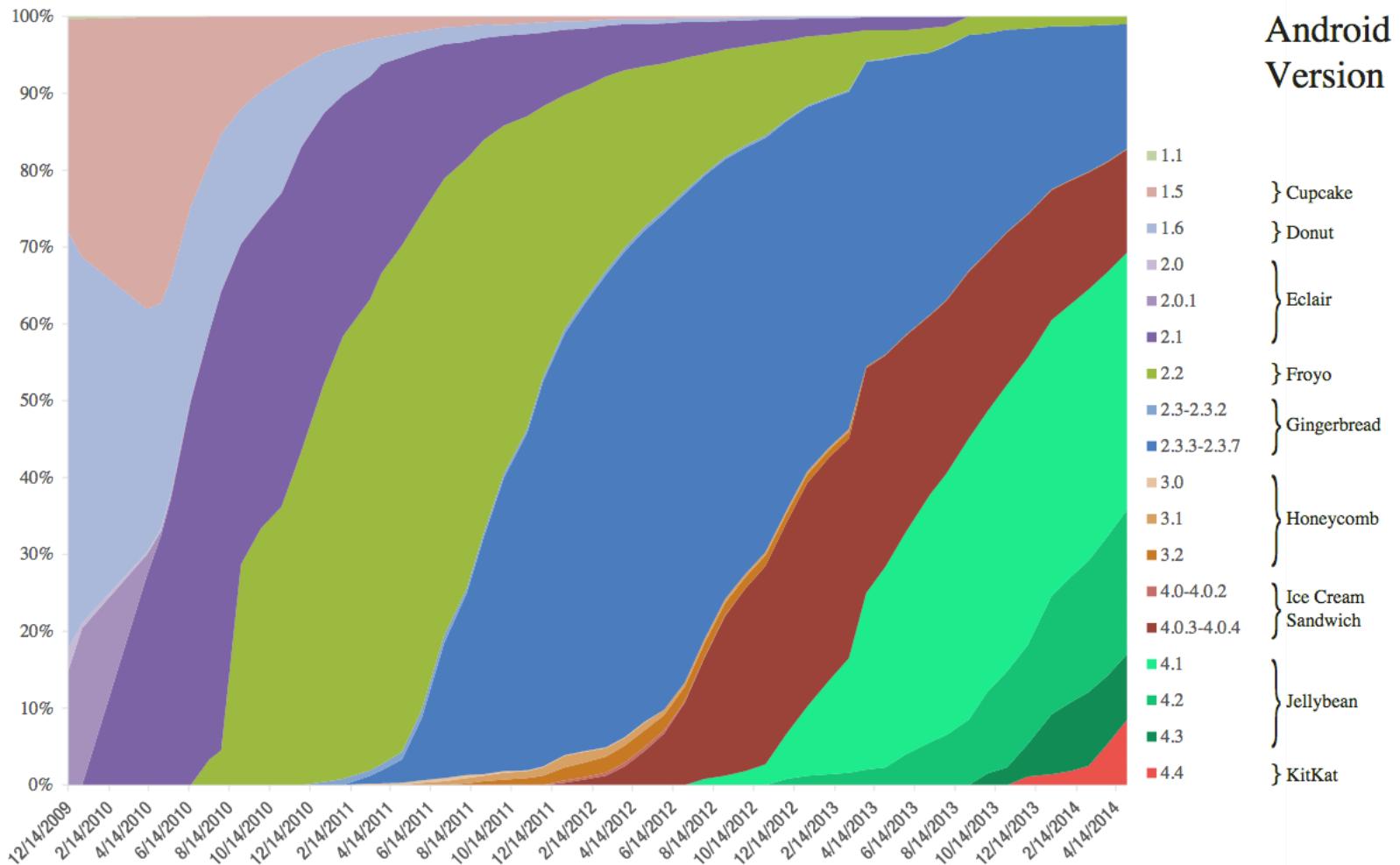


# Historia

## Versiones de Postres



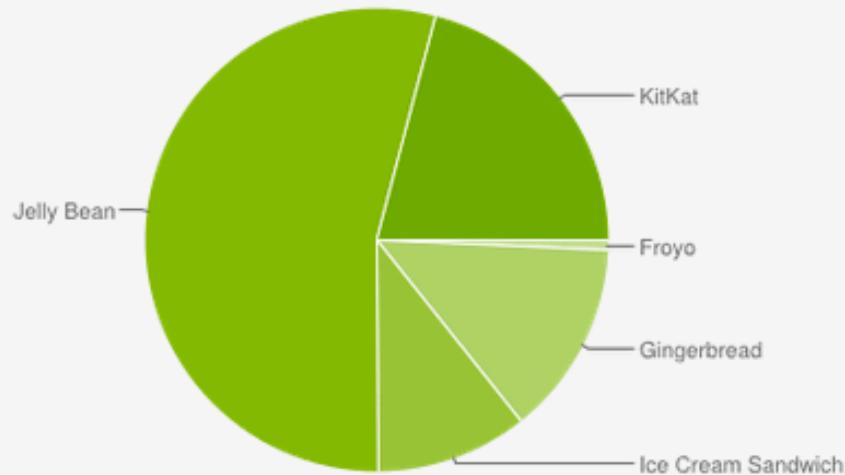
# Historia Mercado de Postres



# [A tener en cuenta...]

## Mercado de Postres

Version	Codename	API	Distribution
2.2	Froyo	8	0.7%
2.3.3 - 2.3.7	Gingerbread	10	13.6%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	10.6%
4.1.x	Jelly Bean	16	26.5%
4.2.x		17	19.8%
4.3		18	7.9%
4.4	KitKat	19	20.9%



Data collected during a 7-day period ending on August 12, 2014.

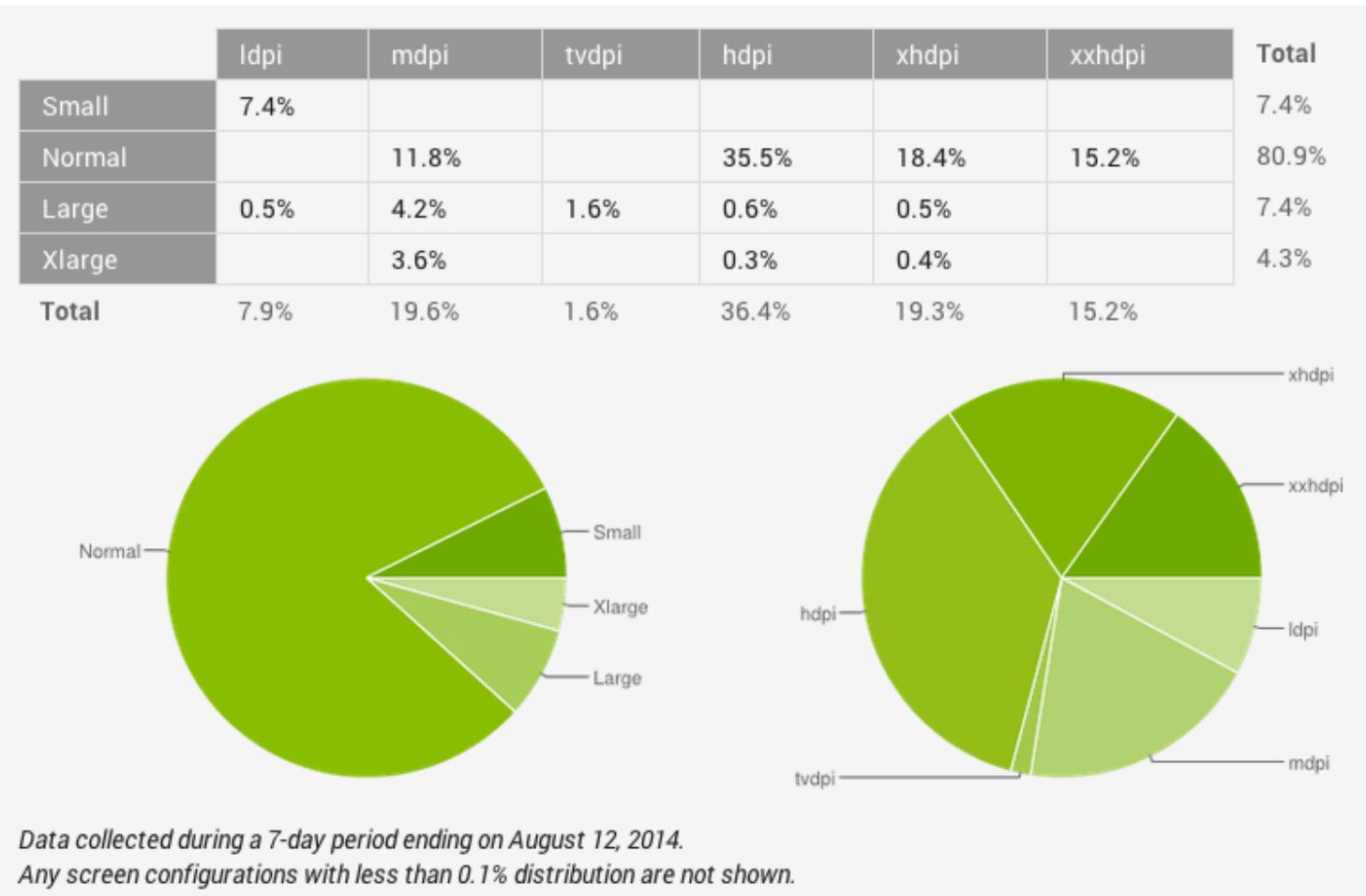
Any versions with less than 0.1% distribution are not shown.

**Note:** Because this data is gathered from the new Google Play Store app, which supports Android 2.2 and above, devices running older versions are not included. However, in August, 2013, versions older than Android 2.2 accounted for about 1% of devices that checked in to Google servers (not those that actually visited Google Play Store).



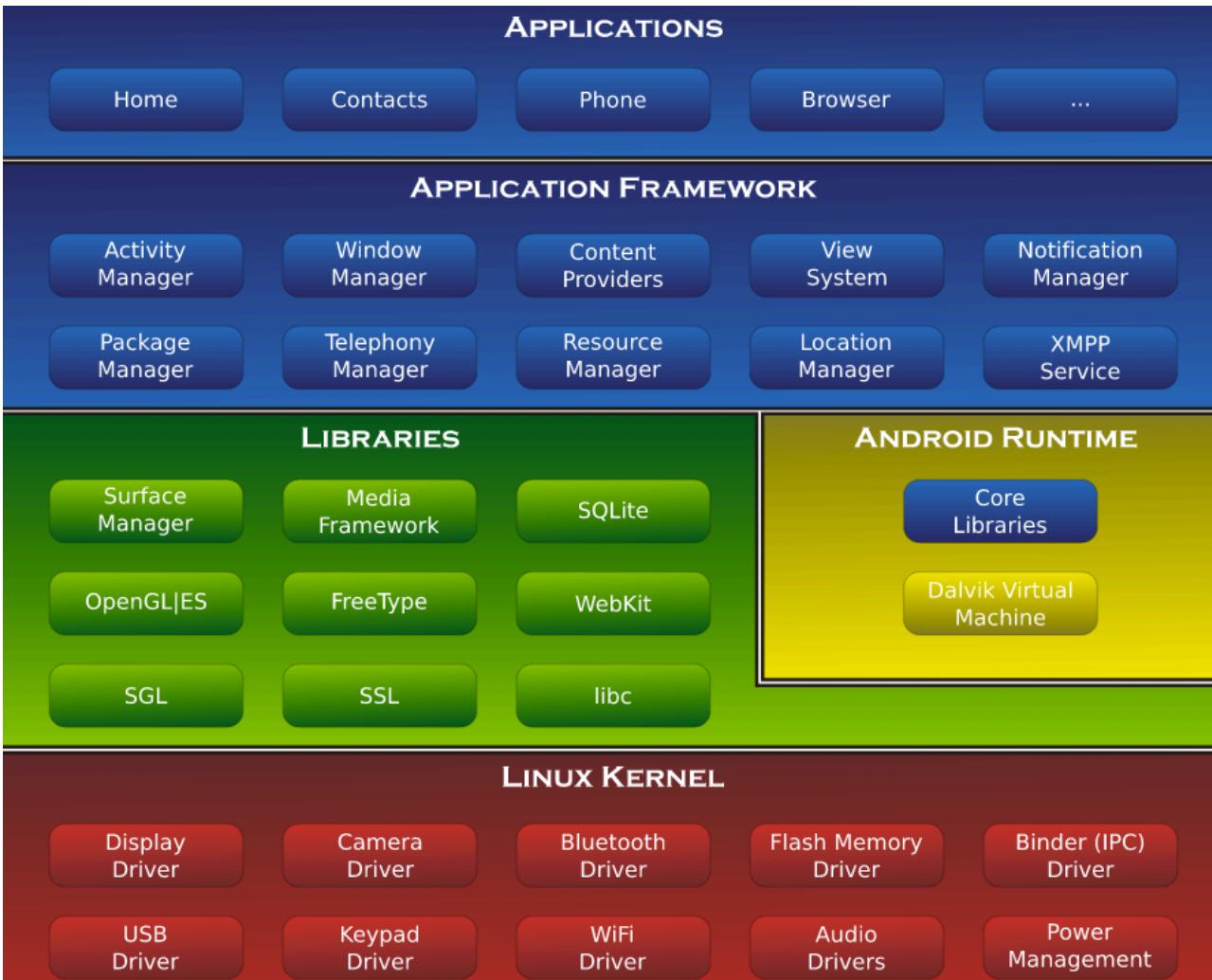
# [A tener en cuenta...]

## Mercado de Pantallas



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

# Arquitectura



# Arquitectura

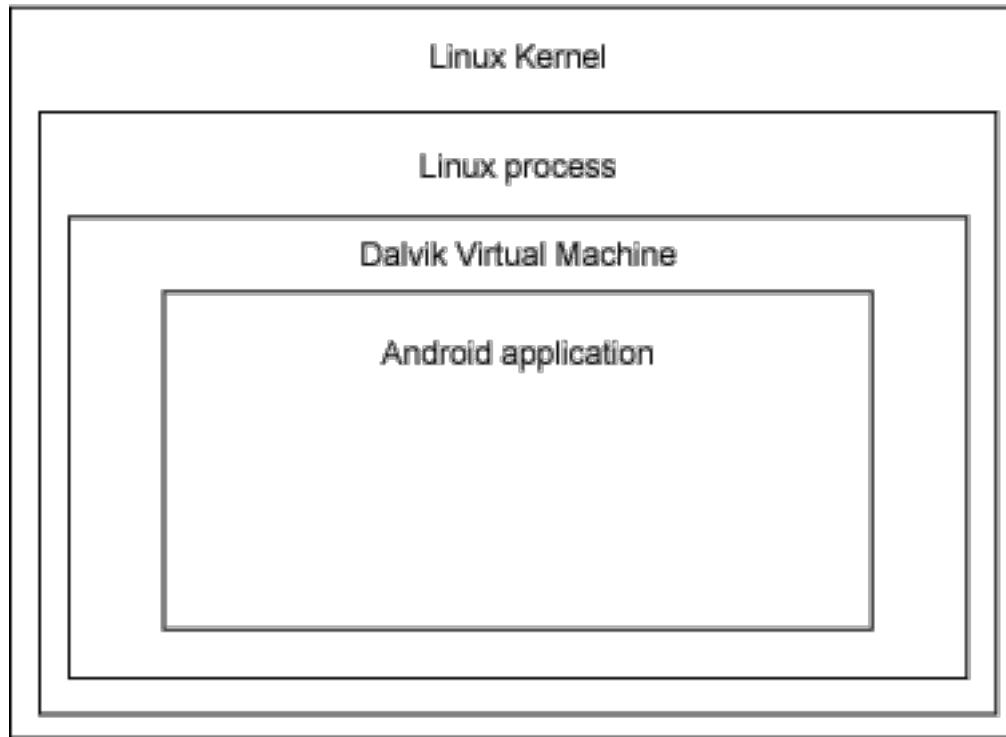
## ¿Máquina Virtual Java?

- Aplicaciones
  - Play Store > Google Play, Android Market
  - Gratuitas o retribución “30% - 70%”
  - Terceros (conocidos o APK)
- Corren “Java” VM (Dalvik DVM)
  - Dan Bornstein
  - Apache (Open Source)
  - Kernel de Linux (drivers)



# Arquitectura

## ¿Máquina Virtual Java?



# Arquitectura

## ¿Máquina Virtual Java?

- ART implementa mejora...
  - Performance
  - Garbage collection
  - Debug y profiling
- Además... Flexycore, Oracle (Sun), ...



# Arquitectura Bibliotecas

---

- FreeType - fuentes de texto
- SQLite - base de datos relacional
- OpenGL - gráficos
- WebKit - navegador
- SSL - cifrado
- SGL - gráficos
- Estándar de C Bionic
- Framework OpenCore



Administrador de interfaz gráfica

# Desarrollo Herramientas

---

- SDK
- IDEs
  - Eclipse
  - Android Studio
  - Netbeans
  - Motodev
  - App Inventor
- Emuladores
- Diseño



# Desarrollo

## Estructura

- /src – Actividades, clases y archivo R.java
  - Clase generada por Android
  - Para acceder a recursos compilados
- /res/drawable – Imágenes
- /res/layout – Vistas en XML
- /res/values – Archivos XML con las variables
- Libraries – Referencia el SDK de Android
- **AndroidManifest.xml** – actividades y config.



# Desarrollo ... y compilación

---

- Aplicaciones en Java + XML
- Compila bytecode para la JVM
- Se traduce a bytecode para DVM
- Se almacena en
  - .dex (Dalvik EXecuable)
  - .odex (Optimized Dalvik EXecuable)
- *just-in-T compilation* (cada vez que ejecuta)
- *ahead-of-T (AOT) compilation* (instalación)



ActivityOne.java

activity\_one.xml

ActivityTwo

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >

    <TextView android:id="@+id/create"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/onCreate" />

    <TextView android:id="@+id/start"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/onStart" />

    <TextView android:id="@+id/resume"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string.onResume" />

    <TextView android:id="@+id/restart"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/onRestart" />

    <Button android:id="@+id/bLaunchActivityTwo"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/button" />

</LinearLayout>
```

public class ActivityOne extends Activity {

```
    private static final String RESTART_KEY = "restart";
    private static final String RESUME_KEY = "resume";
    private static final String START_KEY = "start";
    private static final String CREATE_KEY = "create";

    // String for LogCat documentation
    private final static String TAG = "Lab-ActivityOne";

    // Lifecycle counters

    // TODO:
    // Create counter variables for onCreate(), onRestart(),
    // onResume(), called mCreate, etc.
    // You will need to increment these variables' values whenever
    // corresponding lifecycle methods get called
    private int mCreate;
    private int mStart;
    private int mResume;
    private int mRestart;

    // TODO: Create variables for each of the TextViews, called
    // mTvCreate, etc.
    TextView mTvCreate;
    TextView mTvStart;
    TextView mTvResume;
    TextView mTvRestart;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_one);

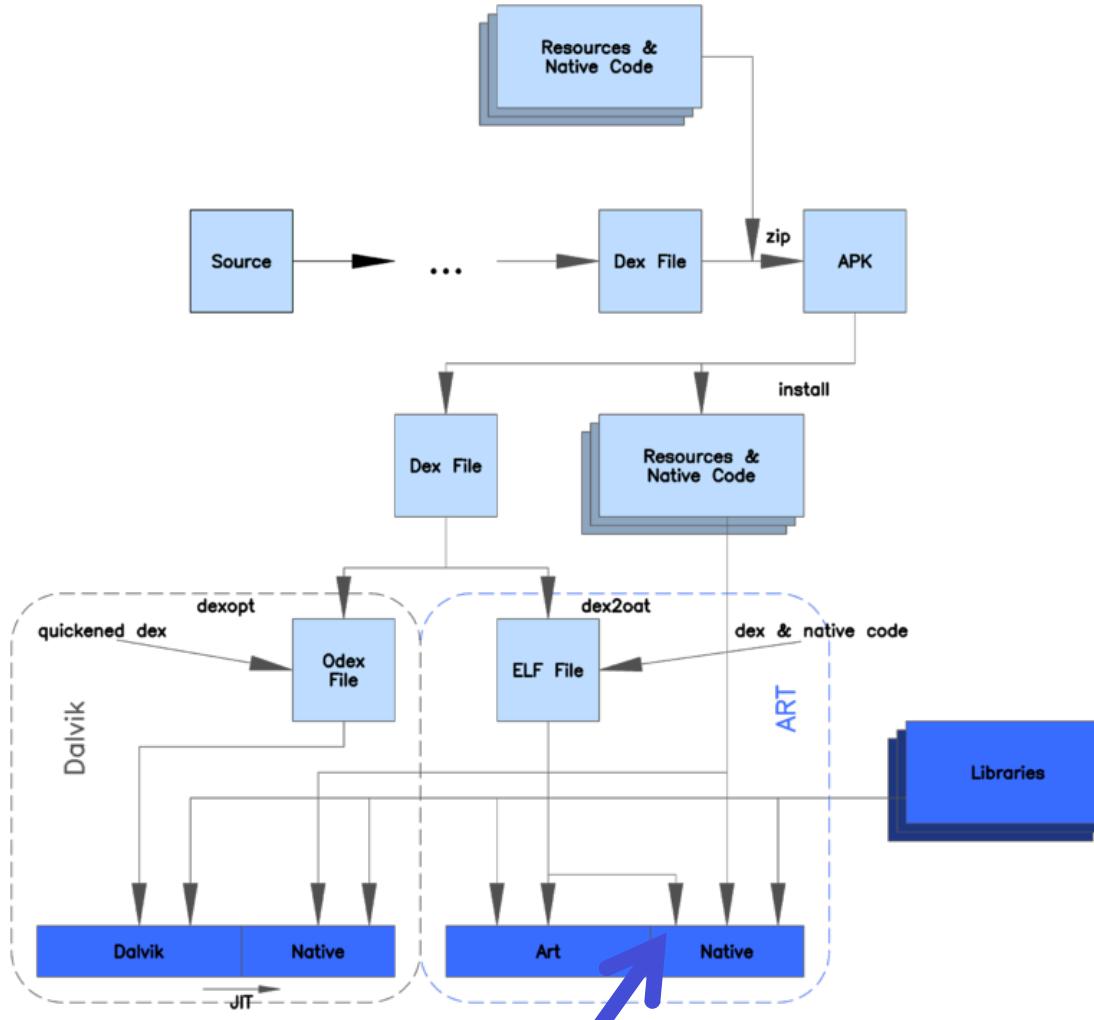
        // TODO: Assign the appropriate TextViews to the variables
        // Hint: Access the views by calling Activity's findViewById()
        // textView1 = (TextView) findViewById(R.id.textView1);
        mTvCreate = (TextView) findViewById(R.id.create);
        mTvStart = (TextView) findViewById(R.id.start);
        mTvResume = (TextView) findViewById(R.id.resume);
        mTvRestart = (TextView) findViewById(R.id.restart);
```

# Desarrollo APK

- Archivo que empaqueta para distribuir
  - Análogo a msi, deb, ...
  - Formato zip basado en JAR
- Código, recursos, certificados, manifest, ...
- application/vnd.android.package-archive



# Desarrollo APK



# Modelo de programación

## Activities

- Se corresponde con una pantalla
  - Tantas activities como pantallas
- Responsable de mantener su estado
  - Ciclo de vida gestionado por el Framework
- Ej. A1 lista de nombres y teléfonos
  - A2 información detallada del contacto
- Definición de interfaz
  - Procedural: actividad con código Java
  - Declarativa: XML como una página HTML



# Modelo de programación

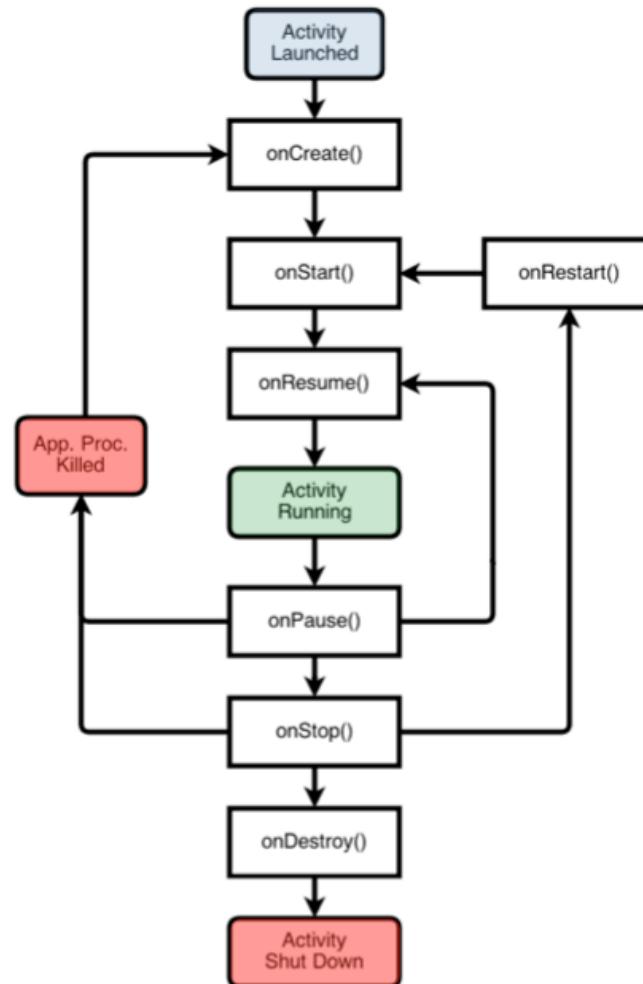
## Activities - Responsabilidades

- Crear una ventana donde colocar la interfaz
- Gestionar la interfaz
- Responder a los eventos (onClick, ...)



# Modelo de programación

## [Ciclo de vida]



# Modelo de programación

## [Procesos]

- Una aplicación ejecuta dentro de un proceso
- El sistema administra la memoria del proceso
- Tiempo de vida controlado por el proceso
- El sistema prioriza los procesos
  - Foreground (Primer plano)
  - Visible (ej. Popup)
  - Service
  - Background
- Empty



# Modelo de programación

## Intents

- Mecanismo para describir una acción
  - Ej. Sacar una foto; enviar un e-mail
- Usos
  - Creamos actividad para el envío de SMS
  - Registraremos para que maneje el Intent
  - Se invocará nuestra actividad



# Modelo de programación

## Services

- Sin interfaz gráfica
- Pueden correr segundo plano (demonio Unix)
- Ejemplos
  - Reproducir un MP3 usando el teléfono
  - Capturar posición GPS avisar amigo cerca
  - Una actividad puede acoplarse para pausarla



# Modelo de programación

## [Broadcast receivers]

- Sin interfaz gráfica
- Detecta y reacciona ante mensajes o eventos



# Modelo de programación

## Contents Providers

- Mecanismo para compartir datos entre apps
- Almacenan datos compartidos entre las apps
- No se muestran detalles de almacenamiento
- Acceso via API del proveedor
- Ej. contactos



# Interfaz de usuario

- Activity
  - Pantalla que muestra al usuario.
  - Compuesta por vistas
- View
  - Controles o widgets
  - Elemento básico que permite interacción
- ViewGroup
  - Composición de vistas
  - Controles complejos y Layout la heredan



# Interfaz de usuario

## android.view.View

- View – Componente básico para la UI
  - Análogo controles de Java o .NET
  - Área rectangular
  - Gestiona el tamaño, dibujado, foco, gestos
  - Texto, Botones, ListView, WebView, ...
  - TextView, EditText, Button, ...
    - Para dibujar textos, botones, fechas, ...
  - Gestión: Listeners; Implementar: Captura



Se pueden extender

# Interfaz de usuario

## Crear ListView

- Diseñar la lista (res/layout/listItem.xml)
- Crear una actividad que herede ListActivity
- Pasarle los datos al adaptador de la lista
- onCreate asignar el ListAdapter a la actividad
- Añadirle la vista a la actividad



# Interfaz de usuario

## Crear WebView

- Diseñar la vista (res/layout/main.xml)
- AndroidManifest.xml (permisos, pantalla)
- Crear actividad
- onCreate llamar SetContentView pasando id



# Interfaz de usuario

## android.view.ViewGroup

- ViewGroup
  - Contiene otras vistas hijas
  - Clase base Layouts y vistas contenedoras
  - Layouts
    - AbsoluteLayout, TableLayout, ...
  - ViewGroups
    - DatePicker, Gallery, GridView, ...



# Interfaz de usuario

## Layout

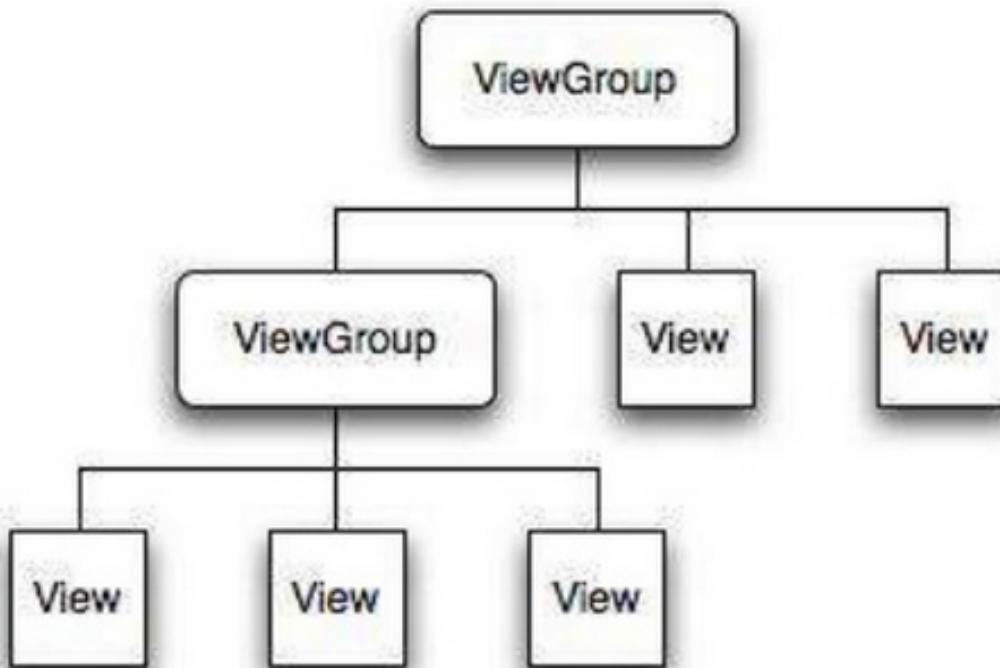
- Contenedores invisibles
- Determinan disposición de vistas en pantalla
- Heredan de ViewGroup
- Contiene LayoutParams especializada



# Interfaz de usuario

## Organización

- Árbol cuya raíz es ViewGroup



- setContentView() permite añadir vistas

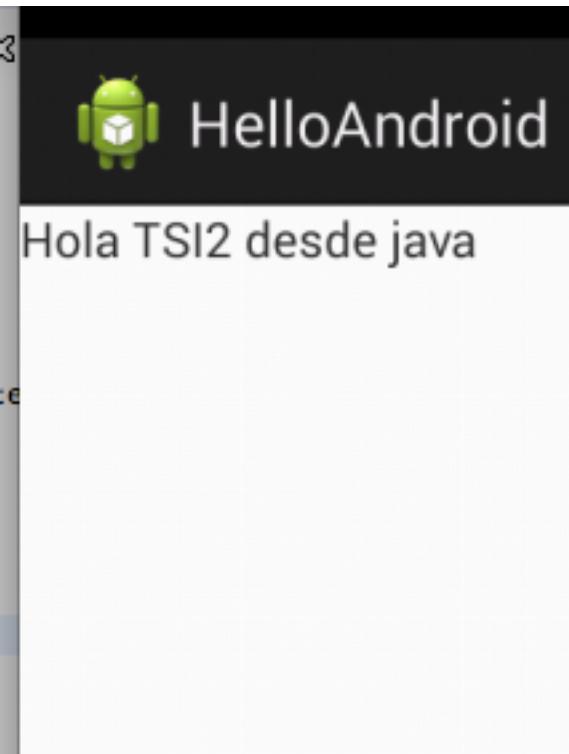


# Interfaz de usuario

## ¿Qué o cómo?

### □ Procedural (Java)

```
activity_main.xml    strings.xml    MainActivity.java
package course.example.helloandroid;
import android.os.Bundle;
public class MainActivity extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        TextView tv = new TextView(this);
        tv.setWidth(100); tv.setHeight(60);
        tv.setText("Hola TSI2 desde java");
        setContentView(tv);
        //setContentView(R.layout.activity_main);
    }
}
```



# Interfaz de usuario

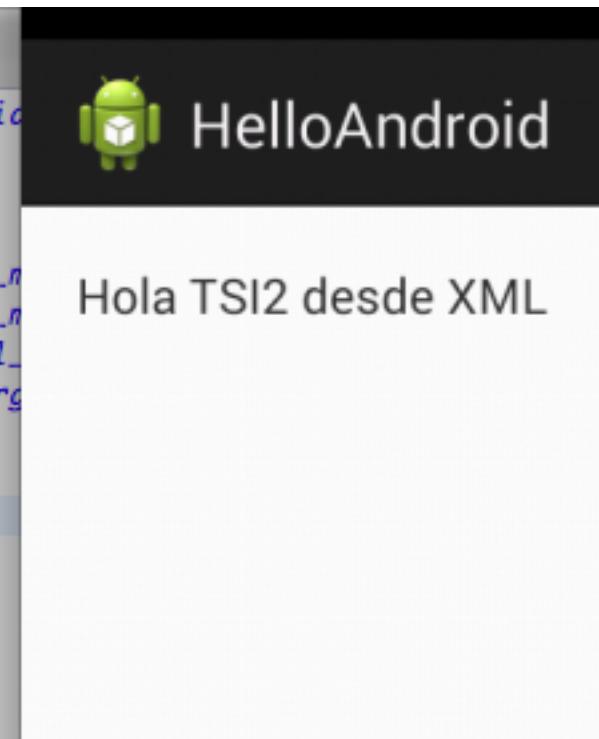
## ¿Qué o cómo?

### □ Declarativa (XML)

```
activity_main.xml strings.xml

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >

    <TextView
        android:id="@+id/nameLabel"
        android:text="@string/hola_tsi"
        android:layout_width="1000px"
        android:layout_height="600px"/>
```



# Interfaz de usuario

## Declarativa

- Separación logica-interfaz favorece MVC
- Modificar interfaz sin modificar el fuente
- id (atributo) identifica la vista dentro del árbol
  - Permite recuperarla desde la app
  - findViewById para acceder desde activity
- @ indica que trate como id recurso
- + indica nombre nuevo → añadir en clase R
- La independencia no es simple...



# Interfaz de usuario

## MVP – Model View Presenter

- P intermediario entre V y M
  - Recupera datos y devuelve formateados
  - Decide sobre la vista
- V implementada por Actividad
  - Referencia a P, V inyecta o crea P
  - Llamará método de P ante cada acción
- M
  - Arquitectura, entrada al dominio, negocio
  - Proveedor de datos



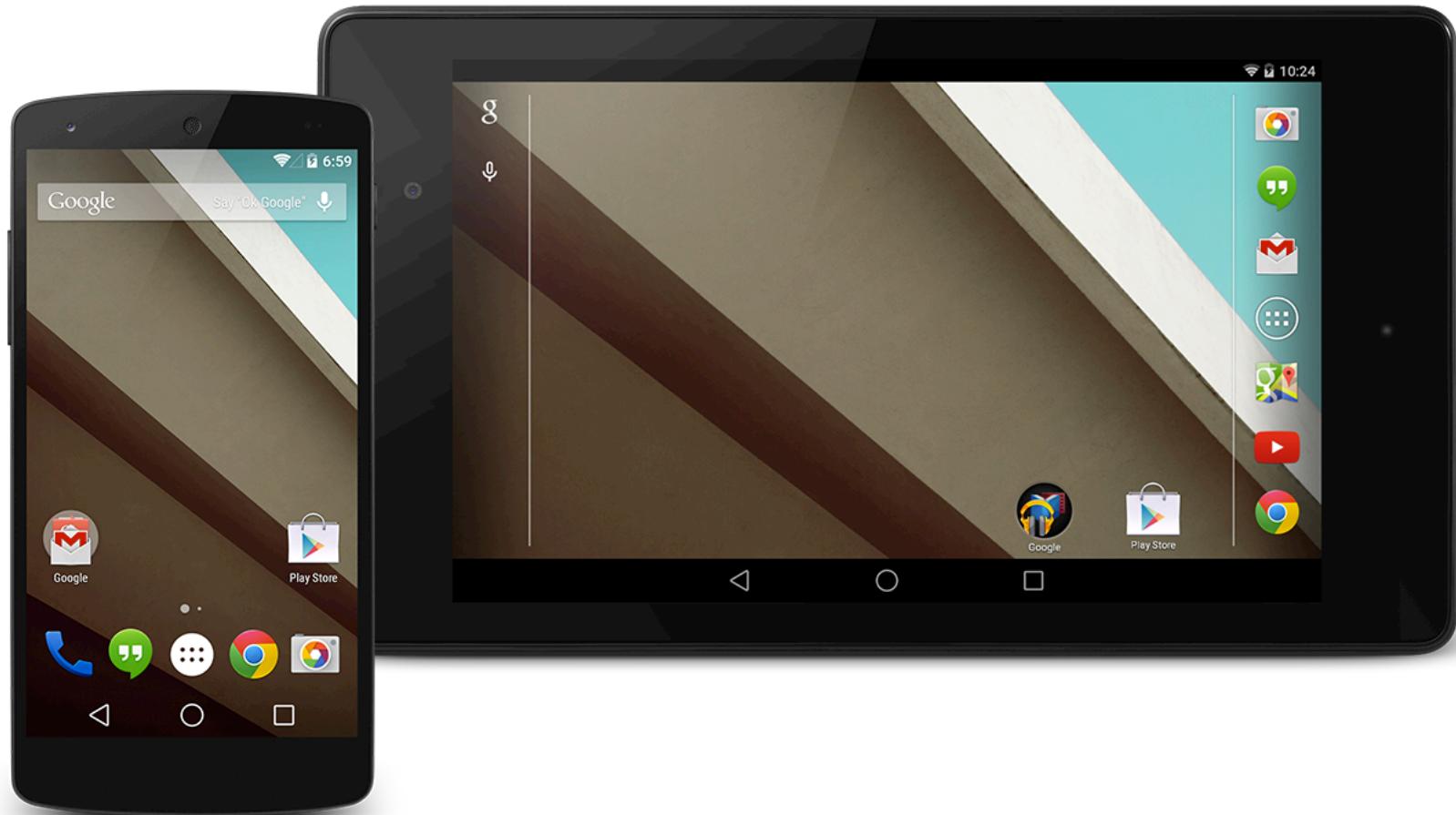
# Testing

## Junit y algo más...

- Se pueden
  - crear casos de prueba JUnit
  - correr como Android JUnit Test
- Cuidado con el *Android Virtual Device*
- Uso de LogCat
- Feedback de usuarios versiones pre-release



# De las últimas... Android “L”



# De las últimas...

## Android “L”

- 2014-06: 25 se da a conocer; 26 beta
- Disponible para “Nexus 5” y “2013 Nexus 7”
- ART remplaza oficialmente a Dalvik
- build LPV79
- Android Open Source Project 2014.07
- build, LPV81C, 2014.08
- Final para fines de este año



# Referencias... y recomendaciones

- Página oficial de desarrollo en Android  
<https://developer.android.com/index.html>
- I, Robot: The Man Behind the Google Phone  
[http://www.nytimes.com/2007/11/04/  
technology/04google.html?  
\\_r=2&pagewanted=all&](http://www.nytimes.com/2007/11/04/technology/04google.html?_r=2&pagewanted=all&)



# Referencias... y recomendaciones

---

- ❑ Ciclo de vida

<http://developer.android.com/training/basics/activity-lifecycle/index.html>

- ❑ Estructura de nuestra aplicación

<http://developer.android.com/tools/projects/index.html>



# Referencias... y recomendaciones

---

- ❑ Interfaz de usuario

<http://developer.android.com/guide/topics/ui/index.html>

- ❑ App Manifest

<http://developer.android.com/guide/topics/manifest/manifest-intro.html>

- ❑ Testing

[http://developer.android.com/tools/testing/testing\\_android.html](http://developer.android.com/tools/testing/testing_android.html)



# Referencias... y recomendaciones

---

- ❑ Entrenamiento de la web oficial

<http://developer.android.com/training/index.html>

- ❑ Seguir las pautas de diseño

<http://developer.android.com/design/index.html>

- ❑ Dudas

<http://stackoverflow.com/>

