

Linux Day 2015



Introduzione allo sviluppo per **Android**

Prof. Riccardo Crosato
Dipartimento di Informatica
I.T.I. "V. E. Marzotto" - Valdagno
android@itismarzotto.it



Android

- Android è un sistema operativo per dispositivi mobili inizialmente sviluppato dalla società Android della Silicon Valley, fondata a fine 2003 da **Andy Rubin**, **Rich Miner**, Nick Sears e Chris White
- Nel 2005 l'azienda viene acquisita da Google con l'obiettivo di fornire una piattaforma mobile completa e aperta
- Il 5 novembre 2007 Google fonda il consorzio **Open Handset Alliance (OHA)** con altre aziende informatiche, presentando pubblicamente la piattaforma Android
- Android diventa una iniziativa open source guidata da Google sotto l'ombrello della Open Handset Alliance



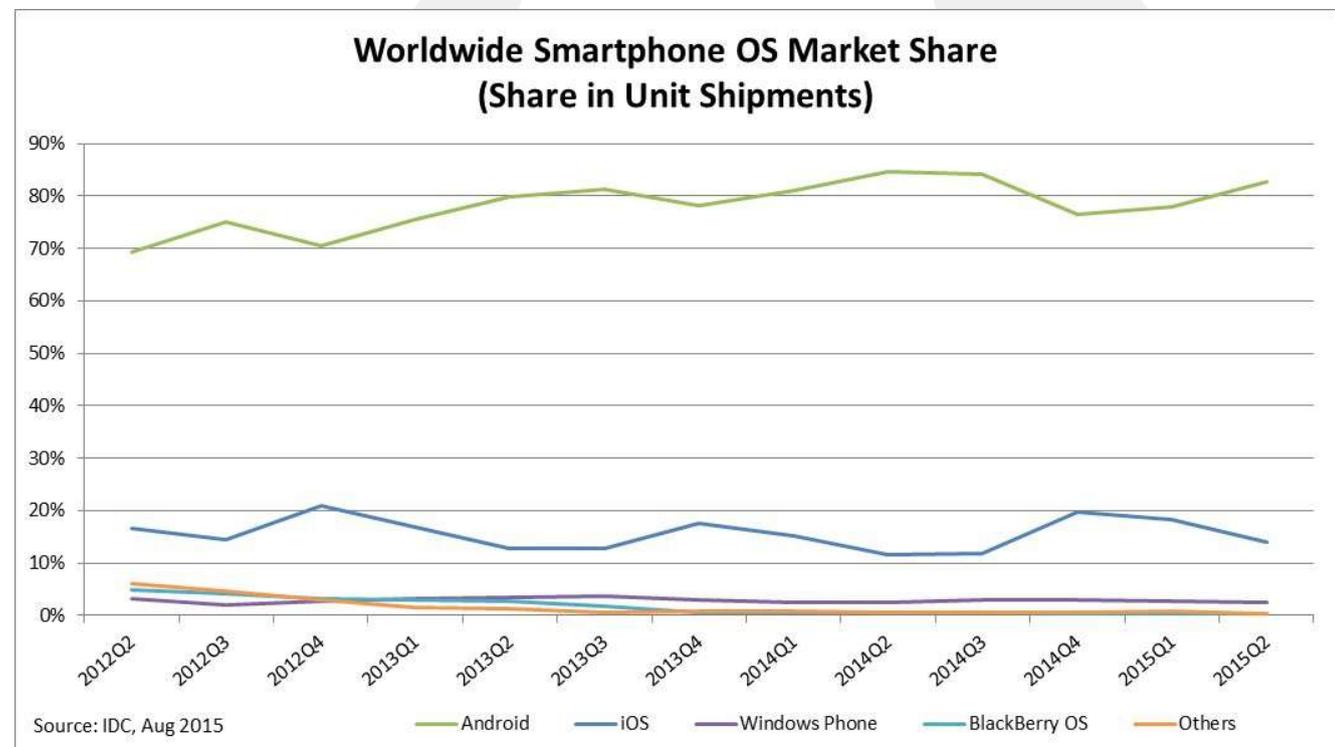


Android – diffusione

- Android è uno dei maggiori attori nel mercato mobile e ha avuto in pochi anni una crescita rapidissima superando aziende leader del settore come Nokia (ora Microsoft), RIM (BlackBerry) e Apple.
- **A livello mondiale Android supera l'80% del mercato** (dato aggiornato ad agosto 2015 - [fonte IDC](#))

Android, oltre ad essere utilizzato per smartphone e tablet, viene sempre più installato su altri dispositivi come

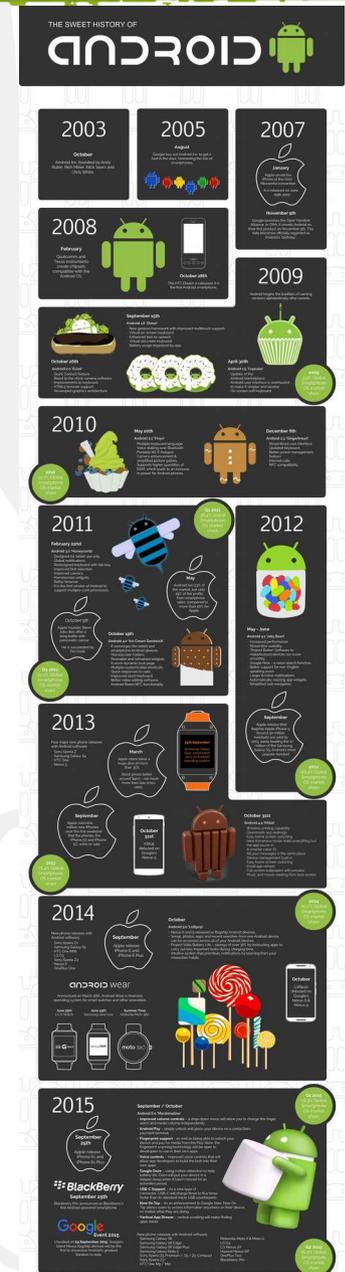
smart TV, autoradio,
fotocamere,
sistemi di domotica,
smart watch, occhiali, ...





Android – versioni e API

- La prima beta di Android è stata rilasciata il 5 novembre **2007**
- Da allora si sono succeduti moltissimi aggiornamenti che oltre a correggere bug hanno aggiunto nuove **API (*Application programming interface*)** al framework
- Per questo motivo oltre al numero di versione di Android è fornito un secondo numero di versione, chiamato livello API (**API level**) che è usato per identificare l'insieme delle API supportate
- Da **aprile 2009** ogni versione di Android è stata rilasciata con un nome in codice (e in ordine alfabetico) basato su nomi di dolci, come **Cupcake**, **Donut** (ciambella), **Éclair** (pasticcino), **Froyo** (yogurt gelato), **Gingerbread** (pan di zenzero), **Honeycomb** (favo di miele), **Ice Cream Sandwich** (biscotto gelato), **Jelly Bean** (caramelle gomgnose) **KitKat**, **Lollipop** (lecca-lecca) e **Marshmallow** (toffolette)
- rif. [Tutta la storia di Android da Cupcake a Marshmallow in una Infografica](#)





Android – versioni e API



1.5

2009/04

Cupcake



1.6

2009/09

Donut



2.0/2.1

2009/10

Eclair



2.2

2010/05

Froyo



2.3

2010/12

Gingerbread



3.0/3.1/3.2

2011/02

Honeycomb



4.0

2011/10

**Ice Cream
Sandwich**



4.1/4.2/4.3

2012/06

Jelly Bean



4.4

2013/09

KitKat



5.0/5.1

2014/11

Lollipop



6.0

2015/10

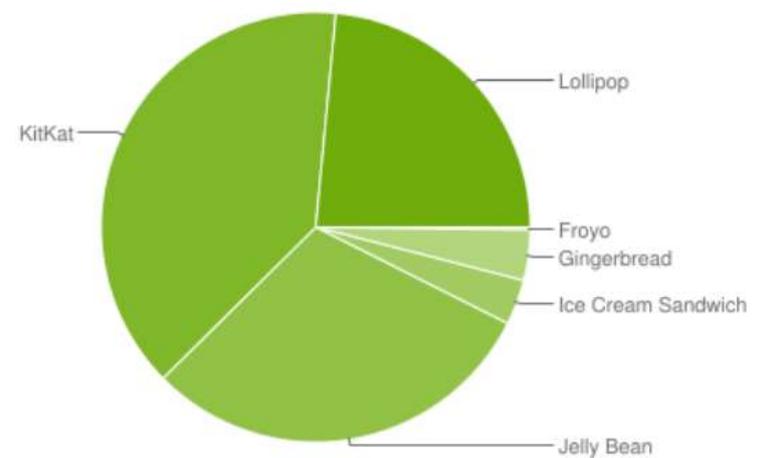
Marshmallow



Android – versioni e API

- <http://developer.android.com/about/dashboards/index.html>
- percentuali di diffusione delle varie versioni – a ottobre 2015 le versioni più diffuse sono la **4.1-4.3 Jelly Bean** (30,2%) e la **4.4 KitKat** (38,9%)

Version	Codename	API	Distribution
2.2	Froyo	8	0.2%
2.3.3 - 2.3.7	Gingerbread	10	3.8%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	3.4%
4.1.x	Jelly Bean	16	11.4%
4.2.x		17	14.5%
4.3		18	4.3%
4.4	KitKat	19	38.9%
5.0	Lollipop	21	15.6%
5.1		22	7.9%



Data collected during a 7-day period ending on October 5, 2015.

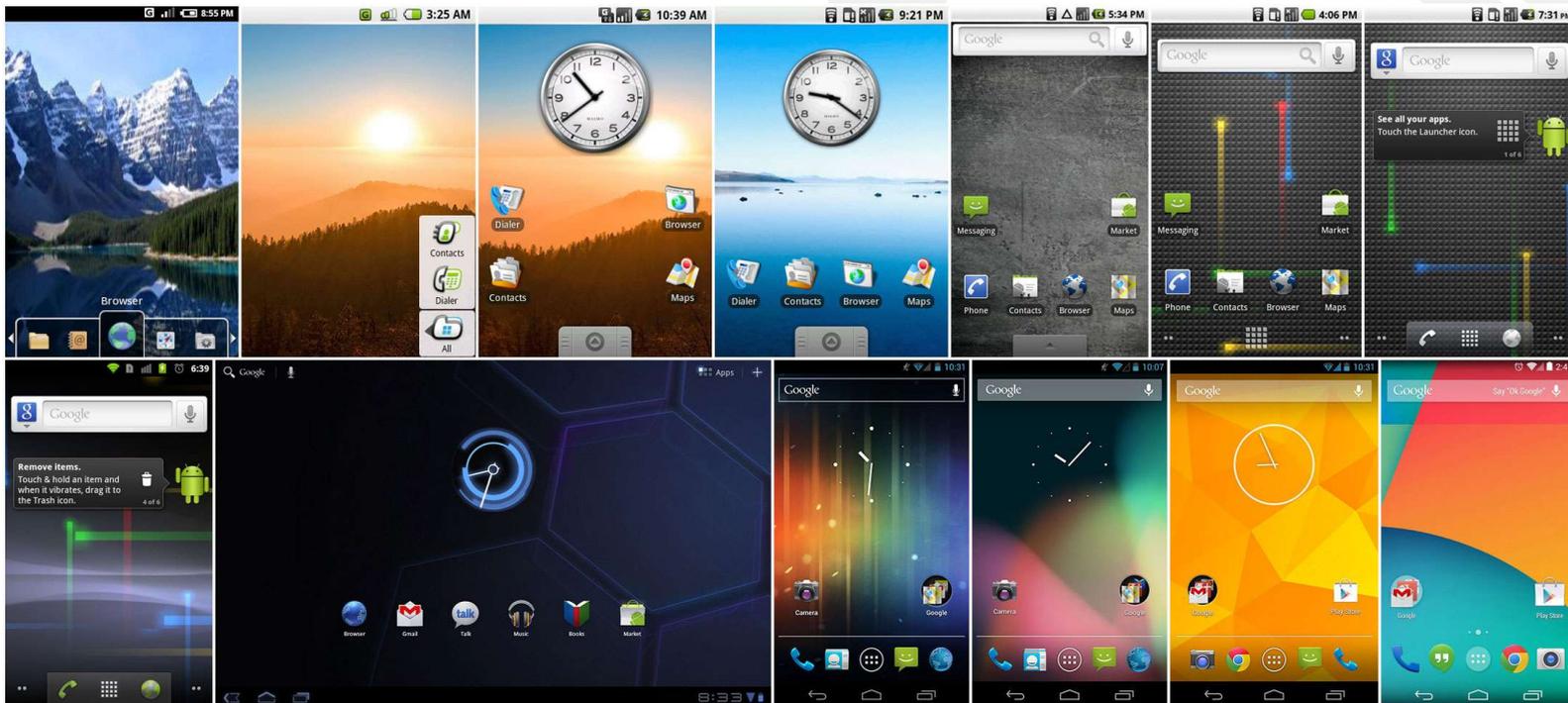
Any versions with less than 0.1% distribution are not shown.



Android – versioni e API

Link sulle differenze tra le varie versioni:

- http://developer.android.com/sdk/api_diff/23/changes.html
(sostituire il numero di API 23 con altre versioni)
- <http://developer.android.com/about/index.html>
- <http://www.androidcentral.com/android-versions>
- <http://socialcompare.com/en/comparison/android-versions-comparison>





Android – componenti

- Android è molto più di un sistema operativo: è la combinazione di strumenti e tecnologie accuratamente ottimizzate per migliorare l'esperienza dell'utente con i dispositivi mobili. Le **principali componenti di Android** sono:



- un **kernel Linux** (versione 2.6 e versione 3.x da Android 4.0 Ice Cream Sandwich) ottimizzato per dispositivi mobili ed embedded che fornisce l'interfaccia a basso livello con l'hardware, la gestione della memoria, il controllo dei processi



open source

- librerie open source per lo sviluppo di applicazioni che includono **SQLite**, **WebKit**, **OpenGL** e gestione dei media 2D e 3D



- un ambiente di esecuzione, sviluppato per essere efficiente e limitare l'utilizzo della memoria, basato su una Java Virtual Machine (**Dalvik Virtual Machine** e dalla versione 5, **ART, Android Runtime**) v. "ART vs Dalvik", articolo dalla Intel Developer Zone



- un framework per le applicazioni per la gestione dei servizi (database, telefonia, sensori, ...) e dell'interfaccia utente



Java

- un kit per lo sviluppo delle applicazioni (**Android SDK, Android Software Development Kit**) che include tutti gli strumenti per lo sviluppo e il debug delle applicazioni oltre ad applicazioni di esempio, documentazione, ...

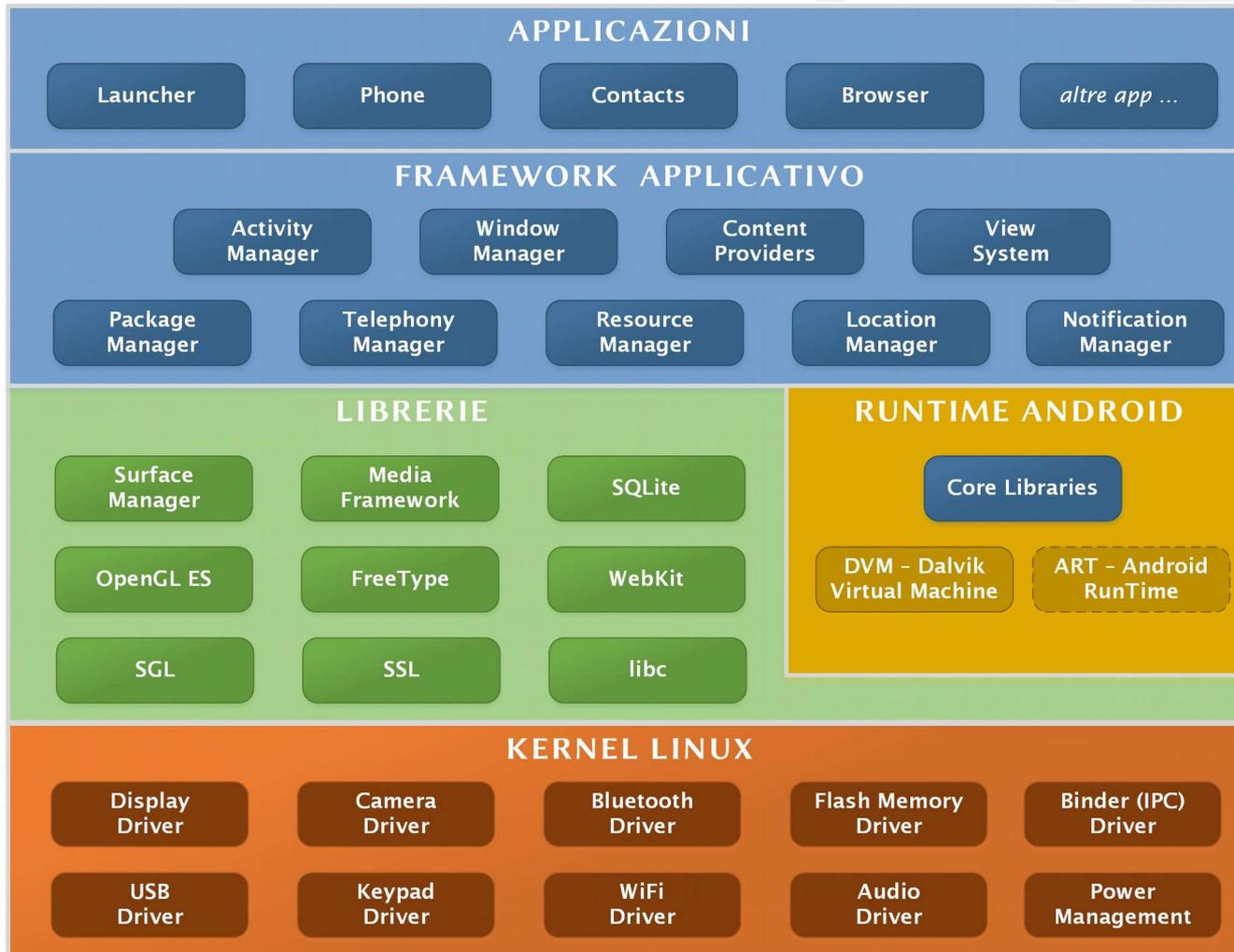


- un insieme di applicazioni preinstallate (client e-mail, gestore della telefonia e degli SMS, rubrica e calendario, browser basato su WebKit, un player audio e video, applicazioni per la gestione della videocamera, calcolatrice, ...)



Android – architettura del sistema

- <http://www.edureka.in/blog/beginners-guide-android-architecture/>
- **Architecture Overview** – tre video dal canale Android Developers





Sviluppo per Android

- Le librerie di runtime di Android (**Core Libraries**) forniscono molte delle funzioni disponibili nelle librerie **Java**
- La scelta degli sviluppatori di Android è stata quella di basarsi su **Java**, ampiamente diffuso e collaudato
- Per esigenze specifiche come il riutilizzo di librerie scritte in **C** e **C++** o più in generale per avere prestazioni più elevate, è possibile usare anche questi linguaggi
 - Allo scopo è disponibile un SDK specifico denominato **Android NDK**, *Native Development Kit*
- **<http://developer.android.com>** è il sito di riferimento per gli sviluppatori Android
- **<http://source.android.com>** è il sito con il codice sorgente e informazioni per costruire dispositivi compatibili con Android



Kit di sviluppo 1 – Eclipse + ADT

1. JDK – Java Development Kit

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

La versione da scaricare è la **Standard Edition** (Java SE)

2. Android SDK – Android Software Development Kit

<http://developer.android.com/tools/sdk/tools-notes.html>

3. Eclipse – ambiente integrato per sviluppare (IDE)

<http://www.eclipse.org/downloads/>

4. ADT - Android Development Tools

<http://developer.android.com/tools/sdk/eclipse-adt.html>

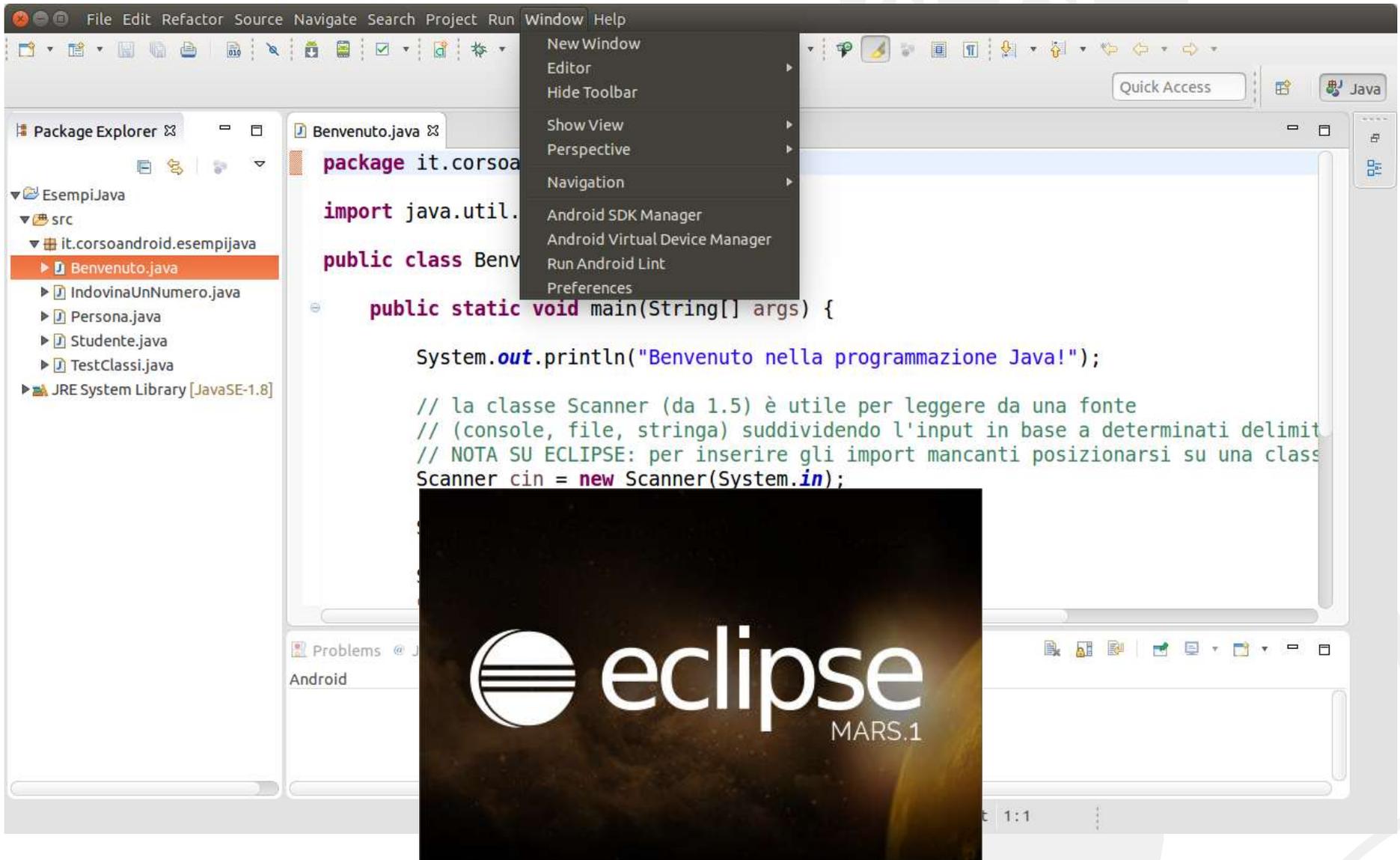
è un plugin da installare in Eclipse (tramite il comando *Help Install New Software...*)

Da fine 2012 a dicembre 2014 veniva distribuito l'**ADT Bundle** che conteneva SDK+Eclipse+ADT già preconfigurati (evitando così i passi 2, 3 e 4)





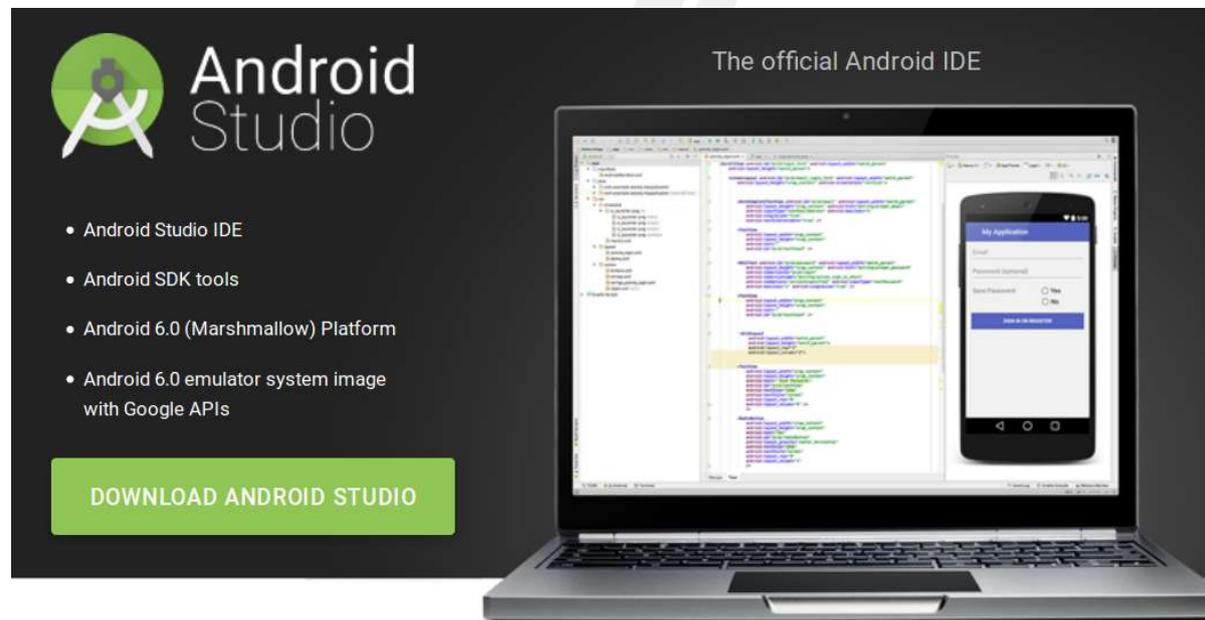
Eclipse – ADT Plugin





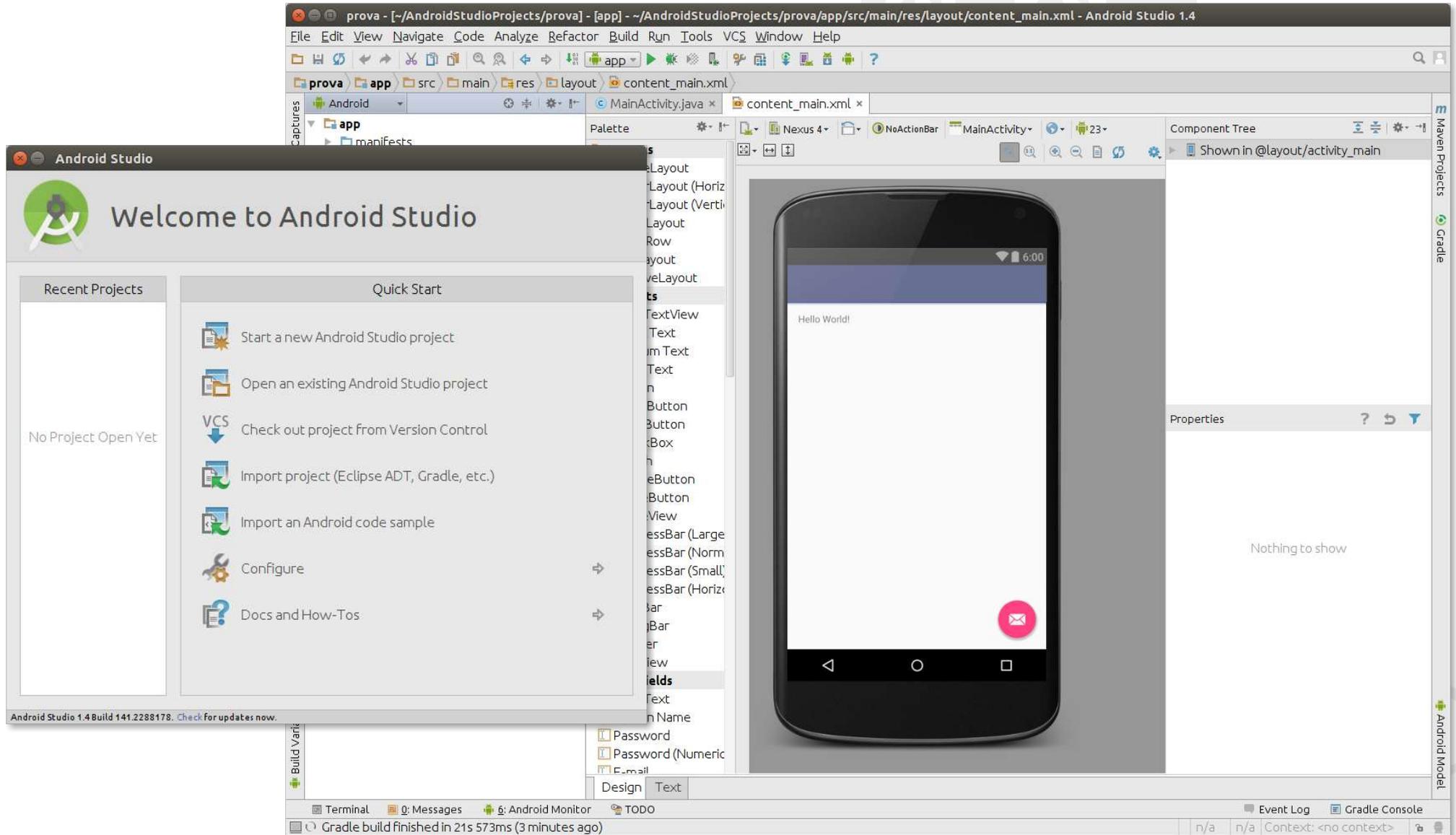
Kit di sviluppo 2 – Android Studio

- Al **Google I/O 2013** (maggio 2013) è stato presentato **Android Studio**, un nuovo IDE alternativo ad Eclipse+ADT. E' basato sull'IDE **IntelliJ IDEA**
- L'8 dicembre 2014 è stata presentata la prima versione stabile 1.0 <http://android-developers.blogspot.it/2014/12/android-studio-10.html> (il supporto all'ADT per Eclipse si è concluso a **giugno 2015**)
- <http://developer.android.com/sdk/index.html>
- **N.B.: va sempre installato prima il JDK**





Android Studio

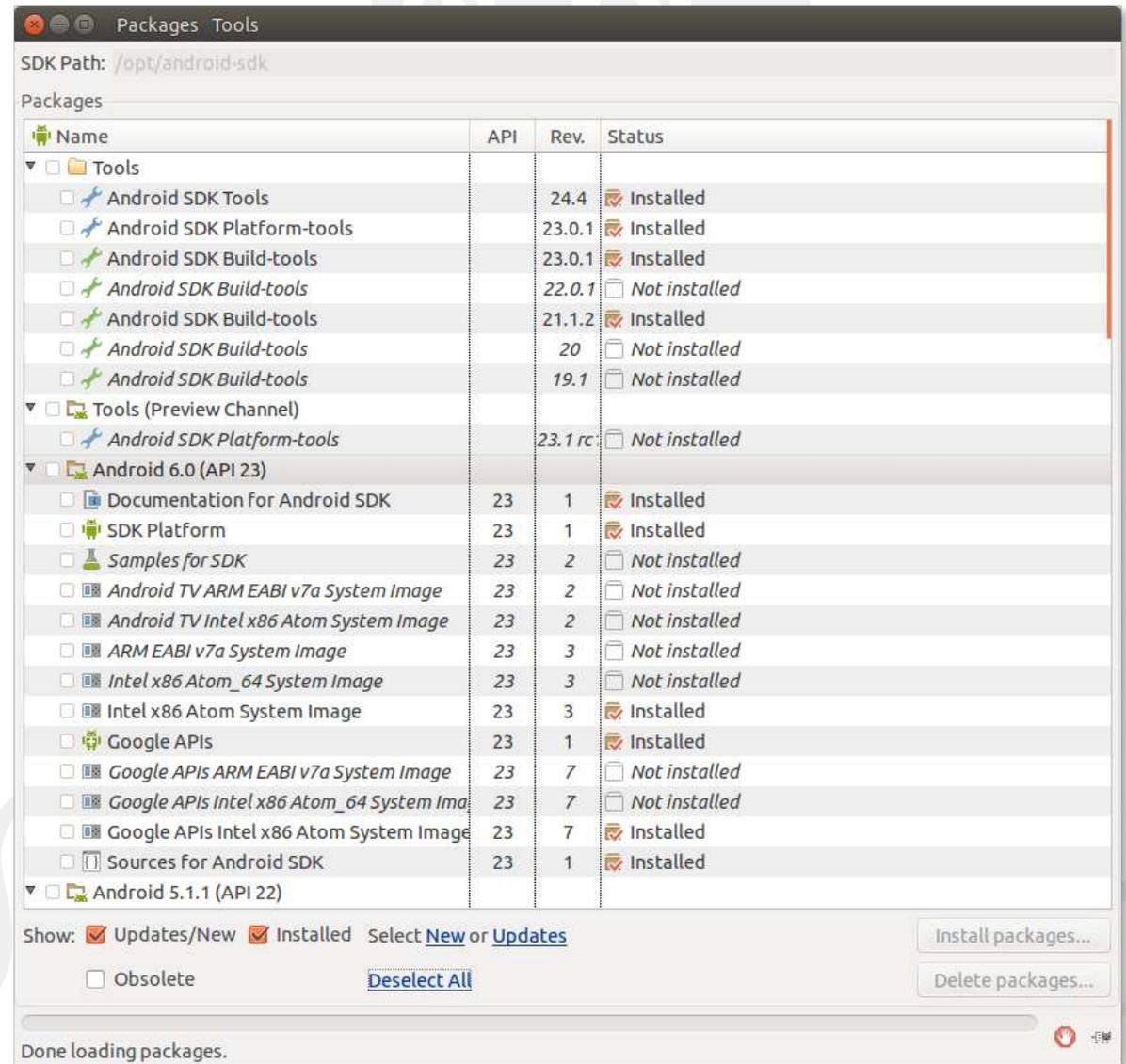




SDK Manager

- Prima di utilizzare Android Studio o Eclipse è opportuno scaricare gli ultimi aggiornamenti, utilizzando **l'SDK Manager**
- Dalla cartella dell'SDK lanciare:

`tools/android`





Android Virtual Device

- L'SDK di Android include un emulatore di dispositivi mobili (**AVD**, **Android Virtual Device**)
- L'emulatore Android simula quasi tutte le caratteristiche hardware e software di un dispositivo reale
 - vengono simulati alcuni sensori → è comunque possibile sfruttare dall'emulatore i dati dei sensori di un dispositivo reale o utilizzare tools esterni come il seguente:
<http://code.google.com/p/openintents/wiki/SensorSimulator>
- Possiamo creare più emulatori da utilizzare per il test delle applicazioni





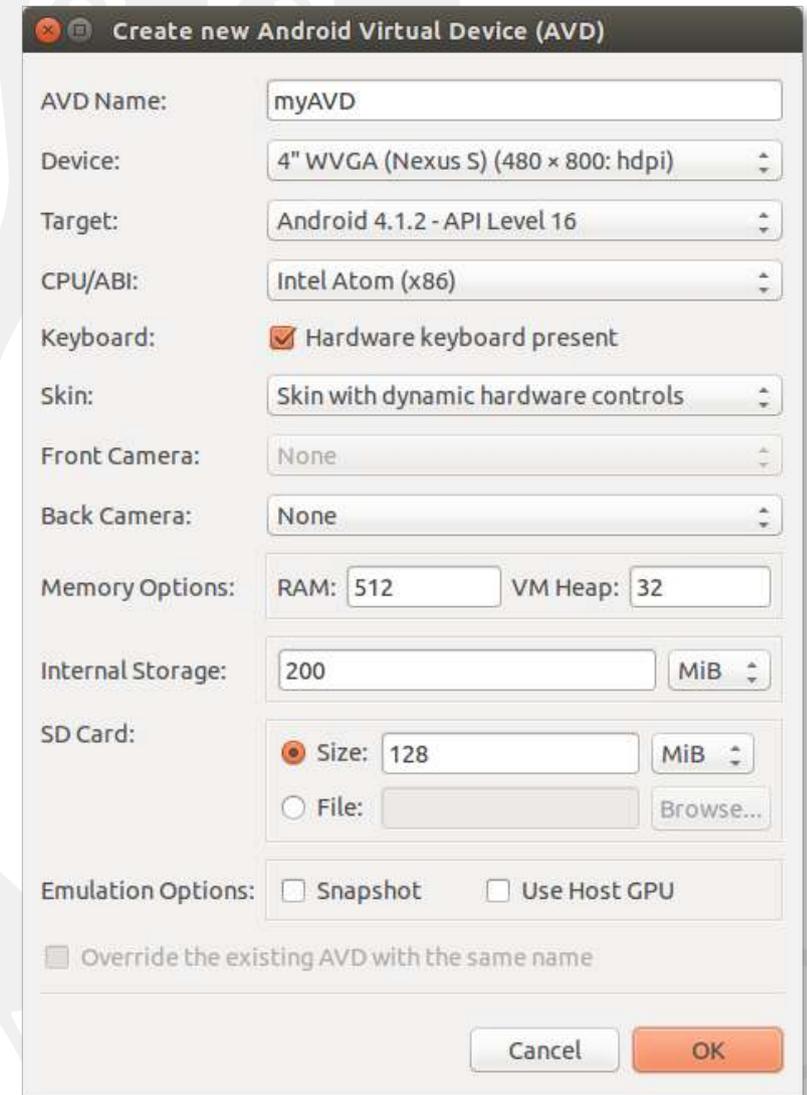
Android Virtual Device

- Per creare un AVD, si può procedere in più modi:
 - dall'**SDK Manager** scegliere dal menu **Tools Manage AVDs...**
 - da **Android Studio**:
Tools – Android – AVD Manager
 - da **Eclipse**:
Window – Android Virtual Device Manager
- Un emulatore può essere creato anche da riga di comando mediante il comando (a partire dalla cartella dell'SDK):

```
tools/android avd
```

dettagli sul comando all'indirizzo:

<http://developer.android.com/tools/devices/managing-avds-cmdline.html>

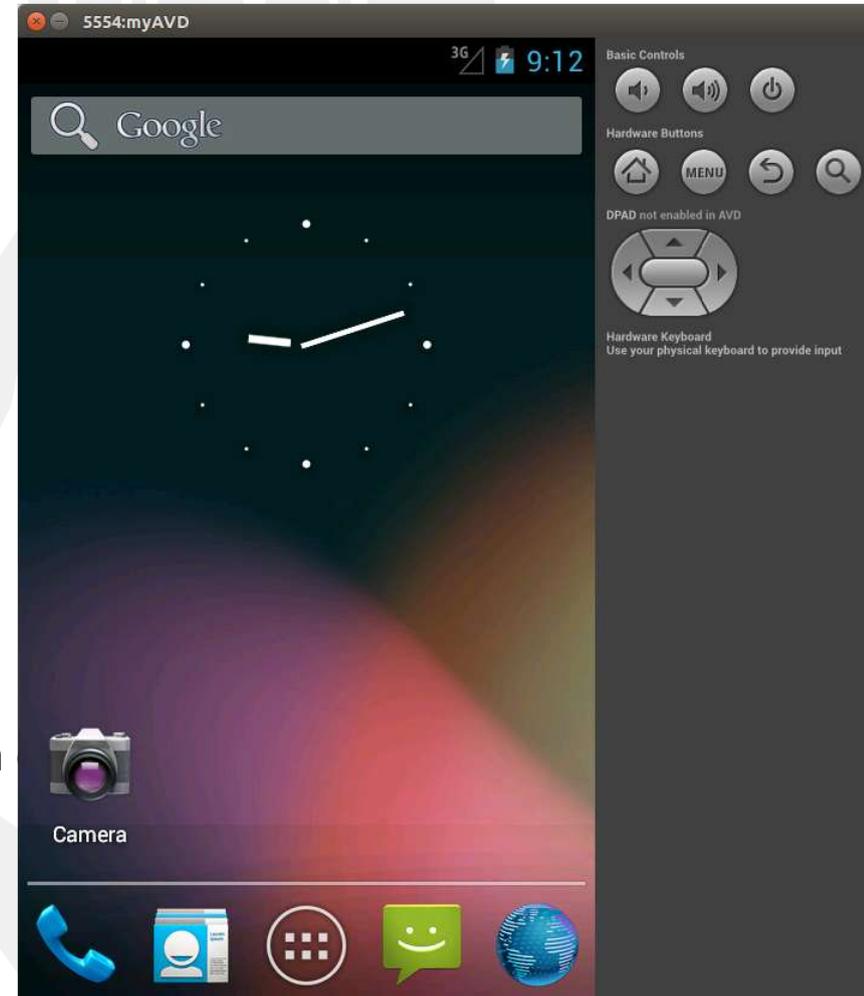
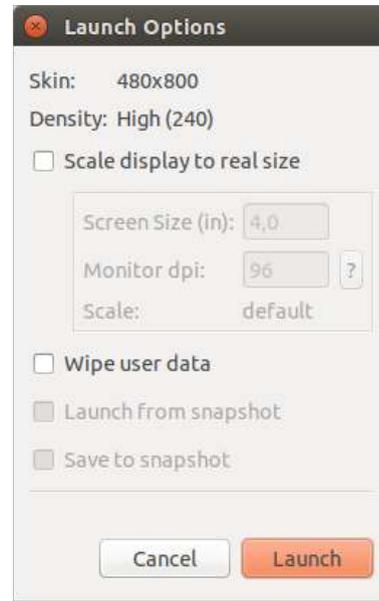




Android Virtual Device

- Un AVD può essere lanciato dall'**Android Virtual Device Manager**

- **Scale display to real size**: ridimensiona il display in base alle dimensioni del video del PC
- **Save to snapshot**: salva un'istantanea per un successivo caricamento più rapido



- Un emulatore può essere lanciato anche da riga mediante il comando:

```
tools/emulator
```

Parametri del comando e utilizzo della tastiera:

<http://developer.android.com/tools/help/emulator.html>



Grazie dell'attenzione
Lascio la parola e la tastiera a

**Harwinder, Pardeep
Luca, Lorenzo e Andrea**