03, 2016.03.01

Android OS/SDK brief intro

Ștefania Ropotică

Index

- Android History
- Android System Arhitecture
- Dalvik application lifecycle
- Memory management

What is Android?

- Android is a software stack for mobile devices that includes
  - operating system,
  - Middleware
  - key applications
- To develop Android application Android SDK is used
- Android SDK include
  - Libraries
  - Development tools
- Documentation on http://developer.android.com

Android History – The beginning

- October 2003, Palo Alto founded Android Inc. In California.
- "smarter mobile devices that are more aware of its owner's location and preferences“
- Google acquired Android Inc. on August 17, 2005
- 12.nov 2007 Android Beta SDK was Released

Android History – Android 1.0

- launched in Sept 2008
- full HTML support in the browser, video playback with a YouTube app, and Google services like GTalk, Gmail, Google Maps and Google Sync.
- Android Market
- Multitasking, Instant Messaging, Wi-Fi Bluetooth
- HTC Dream(G1)

Android History – Cupcake (1.5)

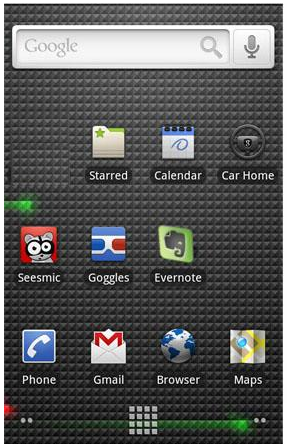
- Linux Kernel 2.6.27
- Soft keyboard
- Faster GPS
- Faster Camera (swap between Still Video)
- Direct Upload to YouTube and Picasa

Android History -1.6 (Donut)

- Lunched in 15 Sept 2009
- Voice search online/offline
- Gallery fully integrated with the Camera apps
- App screenshots in Market
- Multilingual TTS
- Battery usage indicator

Android History 2.0/2.1 (Eclair)

- released in Oct 2009
- Contact sync (multiple acount)
- Bluetooth 2.1
- Support for HTML 5
- Live wall paper
- flash support, Scene Modes, white balance
- Nexus One



Android History 2.2 (Froyo)

- May 2010
- Java V8 engine and JIT compiler
- USB Tethering
- Portable Wi-Fi Hotspot
- Adobe Flash 10.1
- Multiple keyboard languages
- remote wipe

Android History Gingerbread (2.3)

- Dec 2010
- Select/copy paste words
- NFC
- v2.3.7 introduced Google Wallet
- better battery management tools

Android History - 3 (Honeycomb)

- May 2011
- Optimized for tablets
- Browser with tabs
- Support for external keyboards or gamepads
- handwriting prediction for Chinese characters

Android History -4.0 (Ice Cream Sandwich)

- first released on Oct 2011
- drag-and-drop
- Configurable default key
- Face Lock
- Chrome
- Data usage limit
- Panorama view
- Swipte

Android History Android 4.1 – 4.3 Jelly Bean

- June 2012
- Google Now
- Voice dictation is offline
- Faster app updates
- Resizable App Widgets

Android History Android 4.4 KitKat

- Runs on entry-level devices that have as little as 512MB RAM
- NFC capabilities through Host Card Emulation
- Printing framework
- Step Detector and Step Counter
- SMS provider

Android History Android 5.0 Lollipop

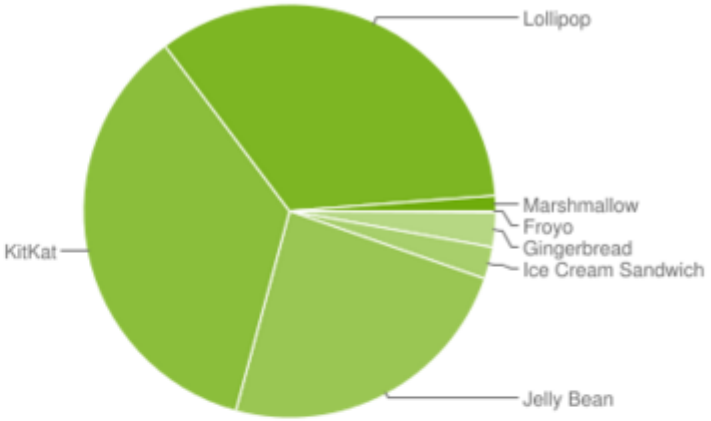
- Material design
- introduces platform support for **64-bit architectures**
- New notifications format
- Support for **Khronos OpenGL ES 3.1**
- New types of sensors:
- **tilt detector**
- **heart rate sensor**

Android History Android 6.0 Marshmallow

- Runtime Permissions
- AudioManager Changes
- Access to Hardware Identifier
- Device connections through the USB port are now set to charge-only
- Camera Service Changes - access model where high-priority processes are favored

Current Distribution

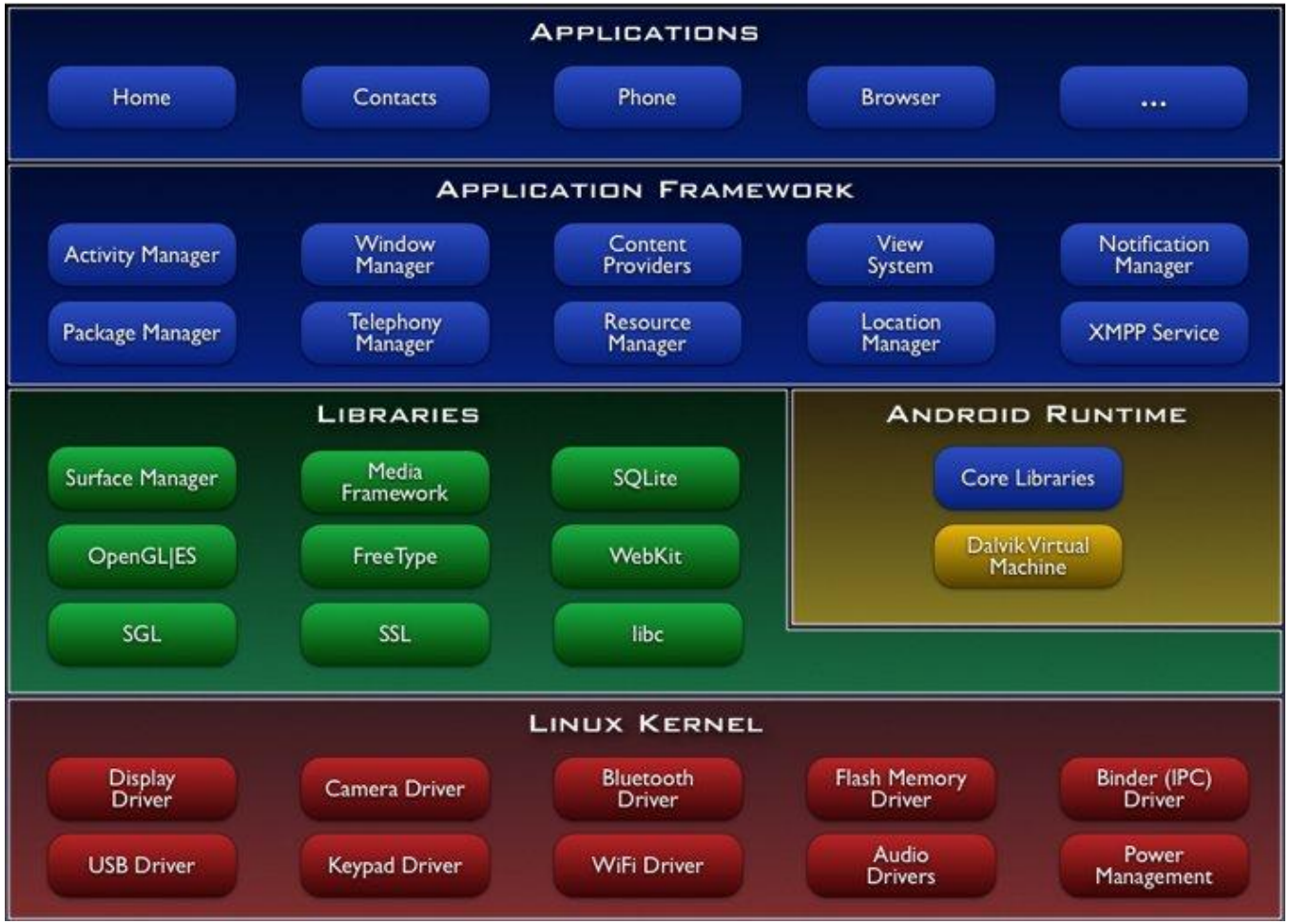
Version	Codename	API	Distribution
2.2	Froyo	8	0.1%
2.3.3 - 2.3.7	Gingerbread	10	2.7%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	2.5%
4.1.x	Jelly Bean	16	8.8%
4.2.x		17	11.7%
4.3		18	3.4%
4.4	KitKat	19	35.5%
5.0	Lollipop	21	17.0%
5.1		22	17.1%
6.0	Marshmallow	23	1.2%



Data collected during a 7-day period ending on February 1, 2016.

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

System architecture

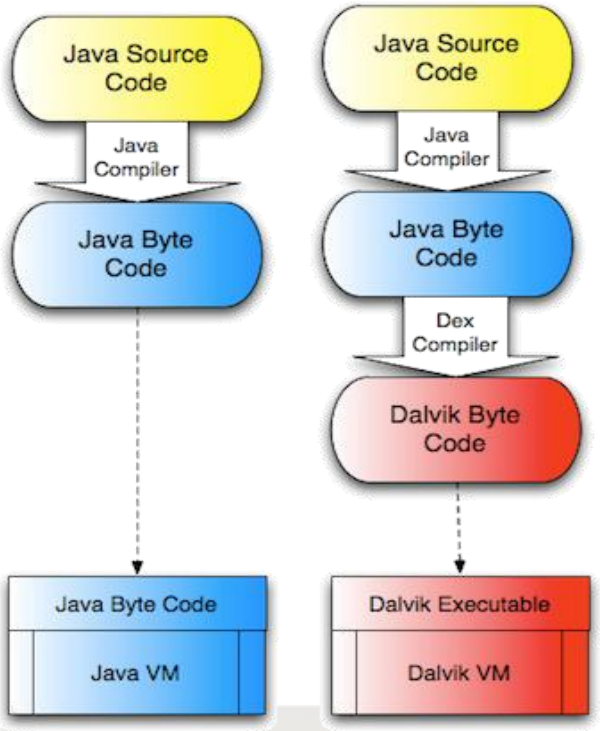


Linux kernel

- Android uses Linux Kernel version 2.6(only the kernel is used from Linux)
- On this layers are the drivers
- This layer takes care of tasks related to :
  - system security,
  - memory management,
  - process management,
  - network stack
  - peripheral (audio, video, GPS, WIFI, Bluetooth..)
  - driver model
- It also acts as an abstraction layer between the hardware and the rest of the software stack.
- This layer is protect by GPL license. Upper levels have Apache License

Runtime Middleware

- VM. Dalvik
  - VM for JAVA optimized for mobile devices
  - Execute only dex code (.java --> .class --> .dex small and fast)
  - 16 bit,register based
  - From 2.3 Dalvik has a just-in-time compiler (JIT)
  - No AWT, Swing or RMI(remote method invocation)
- Core Libraries
  - JAVA APIs: Data Structures, Utilities, File Access, Network Access, Graphics



Native libraries Layer

- (**written in C/C++)**
- **Bionic libc**
  - Google customized C compiler for Android
  - Fast Speed: small size & fast speed of pthread(POSIX Threads) implementation
  - Built-in support for important Android-specific services such as system properties and logging.
- **SQLite**
  - Light-weight relational database management system
- **WebKit web browser**
  - Open source
  - Render pages in full (desktop) view
  - Support Full CSS, JavaScript, DOM, AJAX
- **Surface Manager**
  - Combine 2D and 3D surfaces from multiple applications to frame buffer device
  - Surfaces passed as buffers via Binder IPC calls
  - Can use OpenGL ES and 2D hardware accelerator for its compositions
  - Double-buffering using page-flip
- **FreeType**:
  - Rendering bitmap and vector font

Application Framework Layer

•**Defines component APIs used by the core applications** •**Developers reuse/replace the same components easily.**	
Activity Manager	Manage the lifecycle of (window) applications and provide a common navigation backstack
View System	Components to build applications (e.g., lists, grids, text boxes, buttons, and embedded web browser)
Content Provider	Access data from other applications or to share their own data
Resource Manager	Provide access to non-code resources (e.g., localized string, graphics, and layout files)
Notification Manager	Display customer alerts in the status bar

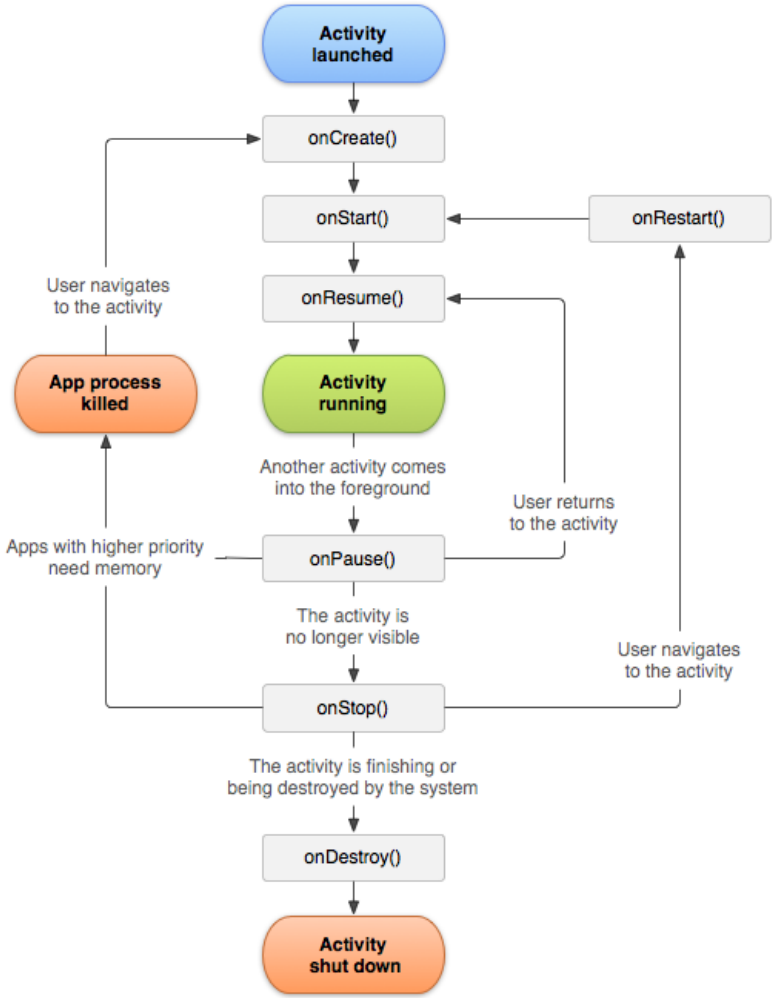
Application Layer

- Written in Java
  - Java .class files must be converted into the .dex format to run on Dalvik
- Core Applications: Email Client, SMS Program, Calendar, Maps, Browser, Contacts

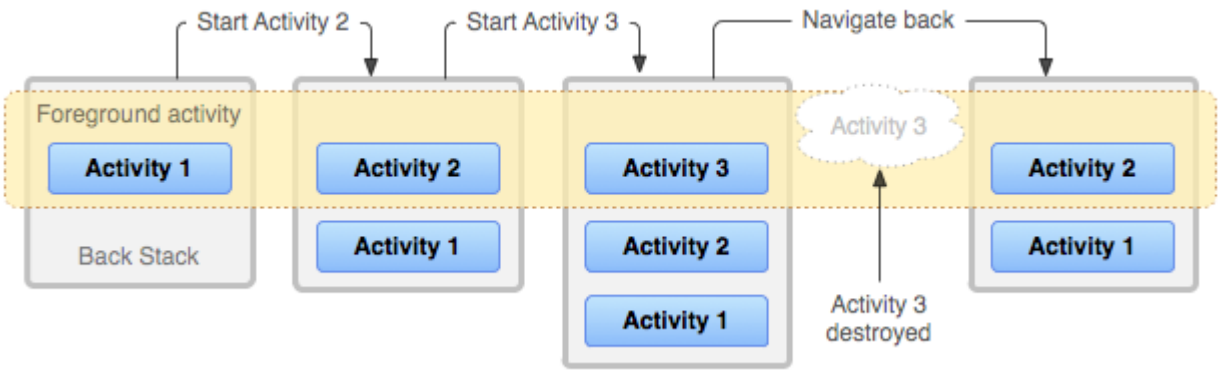
Dalvik application

- Don’t have a **main()**
- Can contains at least one component
  - Activity
  - Service
  - Intent
  - Content provider
  - Sensor Listener

Activity lifecycle



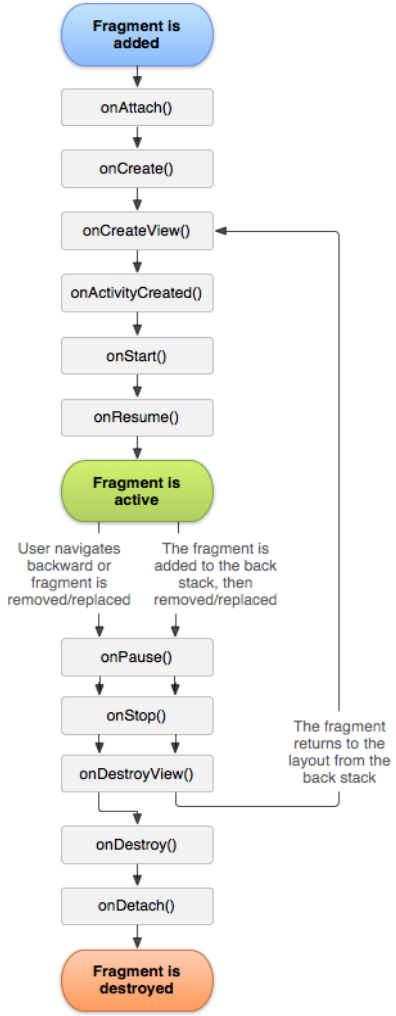
Tasks and Back Stack



*You can—and **should**—proactively retain the state of your activities using callback methods, in case the activity is destroyed and must be recreated*

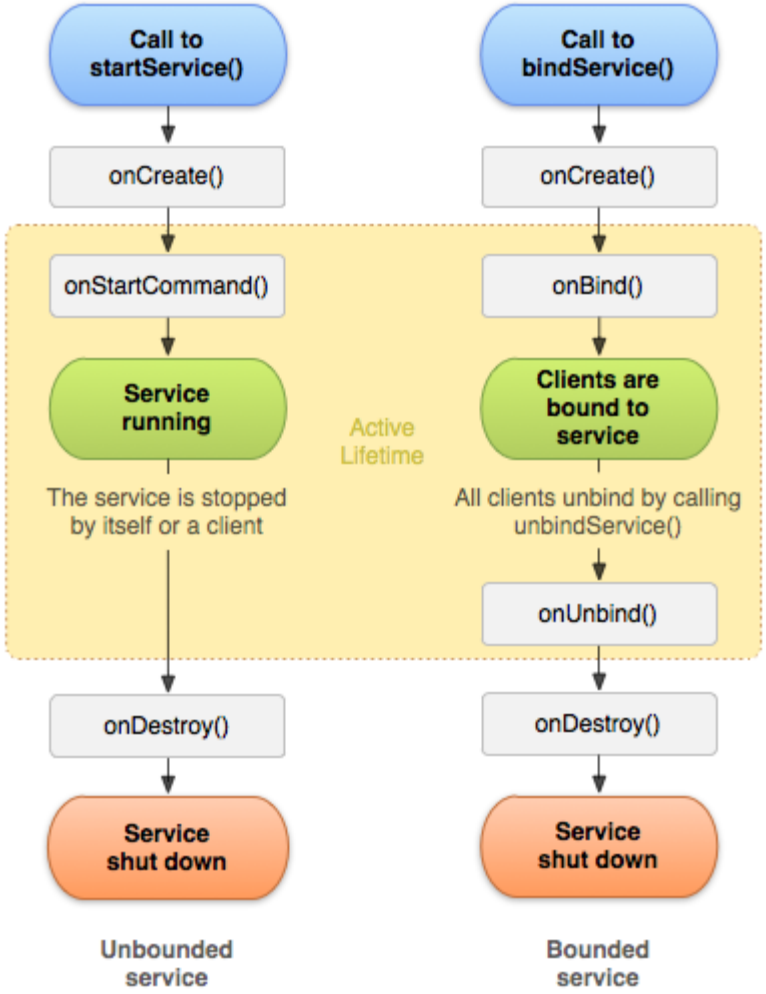
The lifecycle of a fragment

- Can be considered an sub activity of an activity
- Has it's own lifecycle during the lifecycle of the activity that owns it



Services lifecycle

- Used for long operation
- Run in background
- Low priority
- Longer life than activity

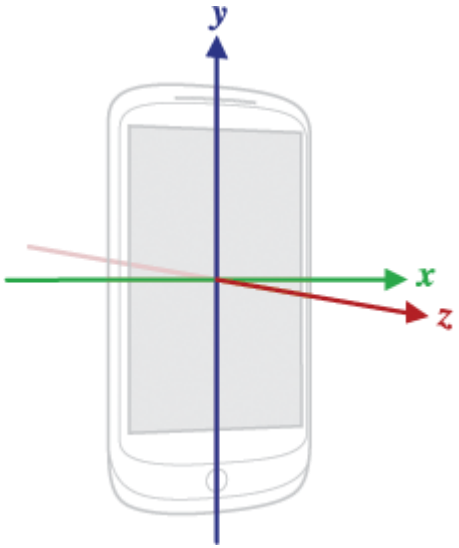


Content Provider

- manages access to a central repository of data
- provide data to other components
- Use SQLite
- Based on URL

Sensor Listener

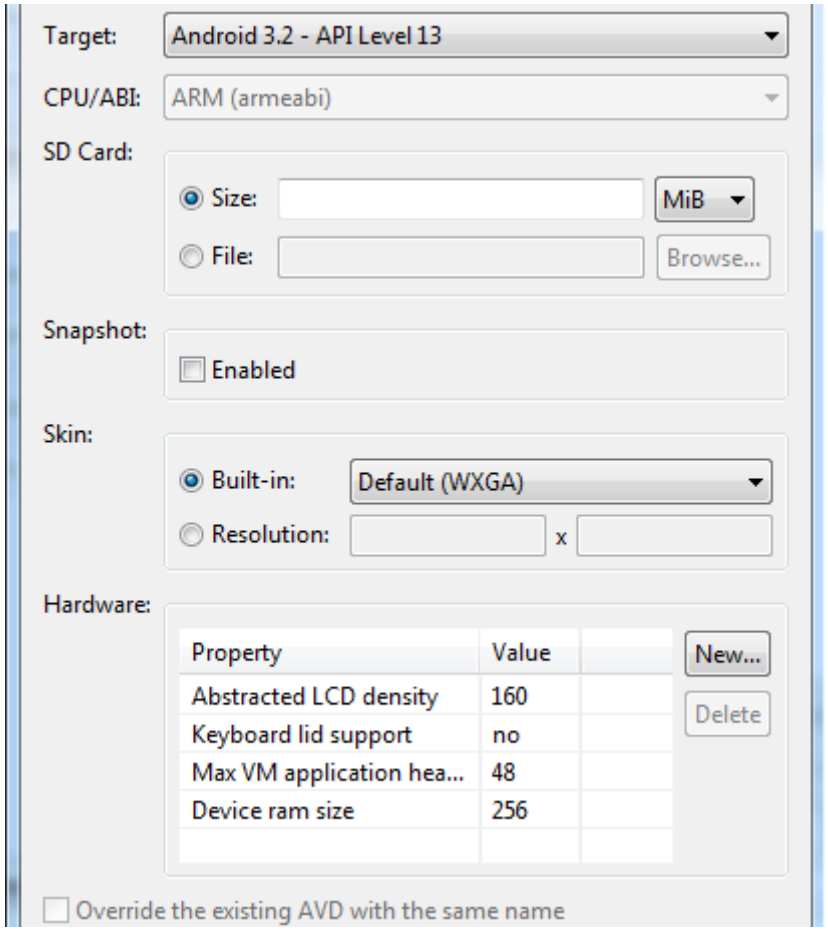
- Motion sensors
  - accelerometers, gravity sensors, gyroscopes, and rotational vector sensors.
- Environmental sensors
  - barometers, photometers, and thermometers
- Position sensors
  - orientation sensors and magnetometers
- Important function
  - registerListener
  - onAccuracyChanged
  - onSensorChanged
  - getSensorList
  - unregisterListener



Memory management

- Heap Size limits
  - G1:16 MB
  - Droid:24 MB
  - Nexus One:32 MB
  - Xoom : 48 MB
- OutOfMemoryError

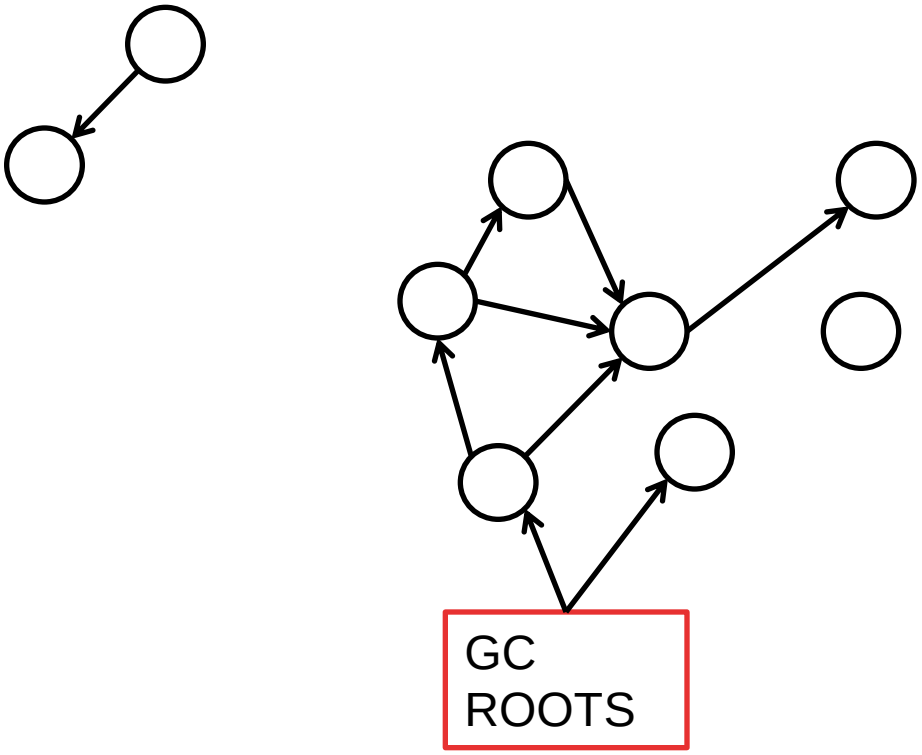
```
<application
...

android:largeHeap="true">

...
</application>
```



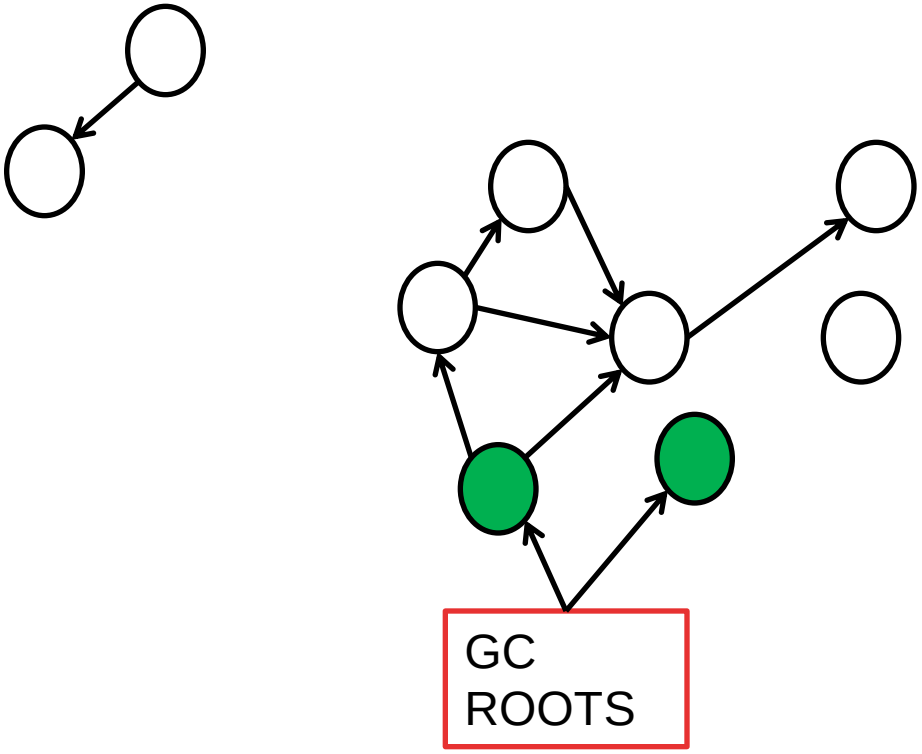
Memory management

- At the shortage of memory the kernel select a low priority process and kill it.
- In Android specification about application life cycle is specified that all application should store their own state

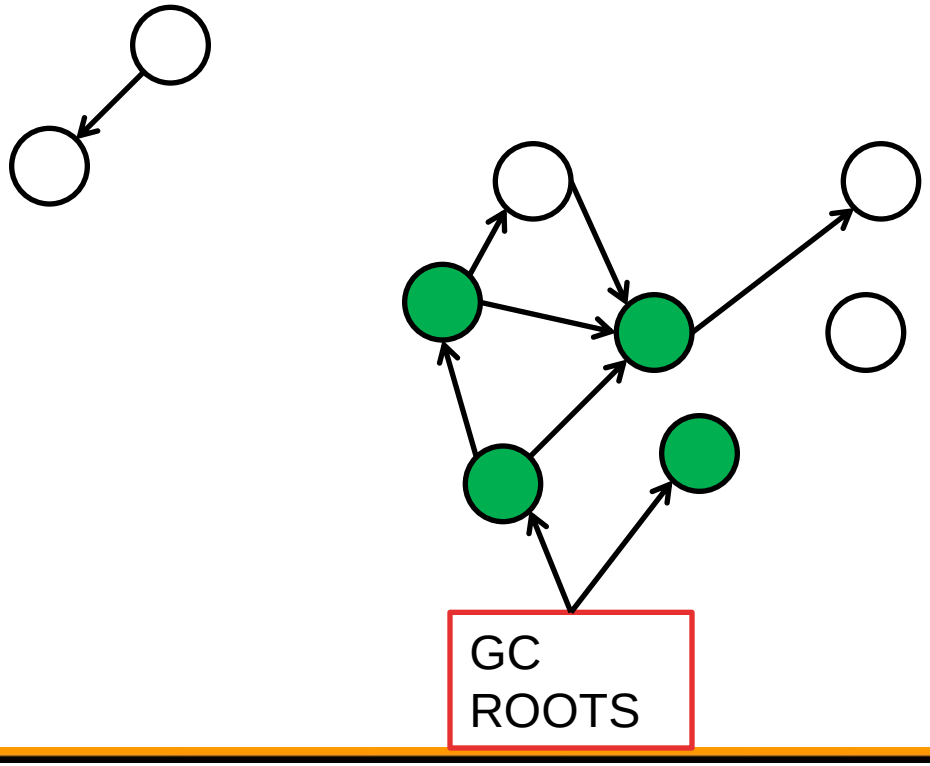
GC



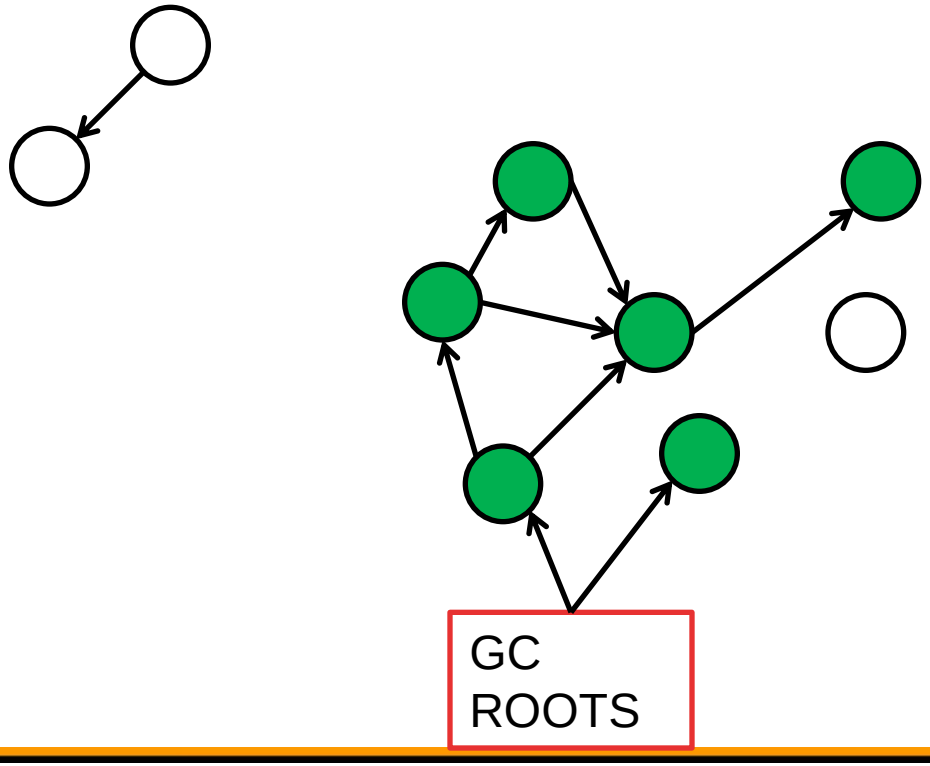
GC



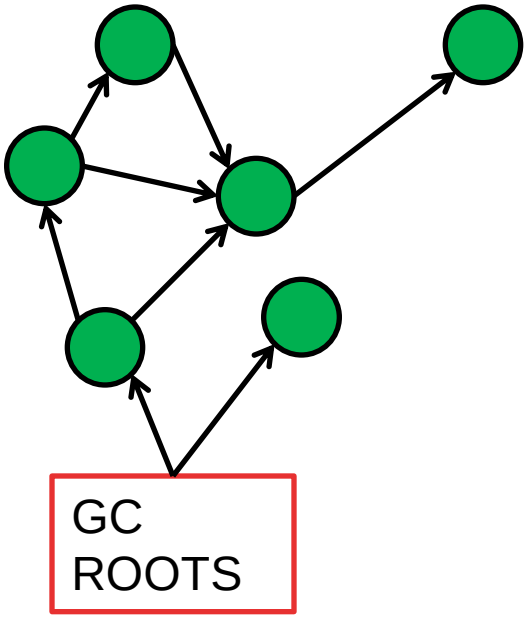
GC



GC



GC

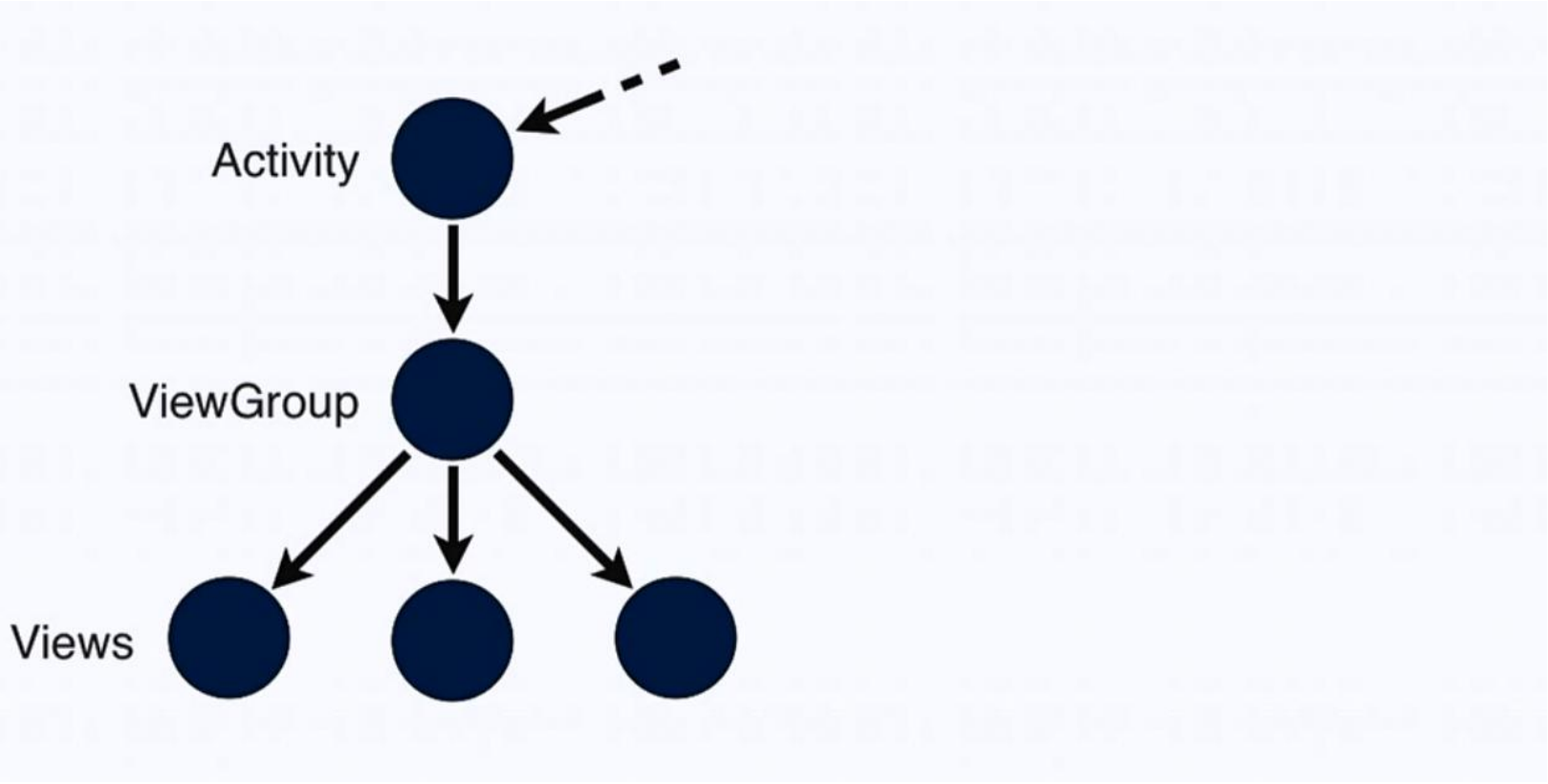


GC improvement

1.0 – 2.2	2.3.x ->
Stop to collect	Concurrent GC
Full heap collection	Partial Collections
Pause often >100ms	Pause time <5ms

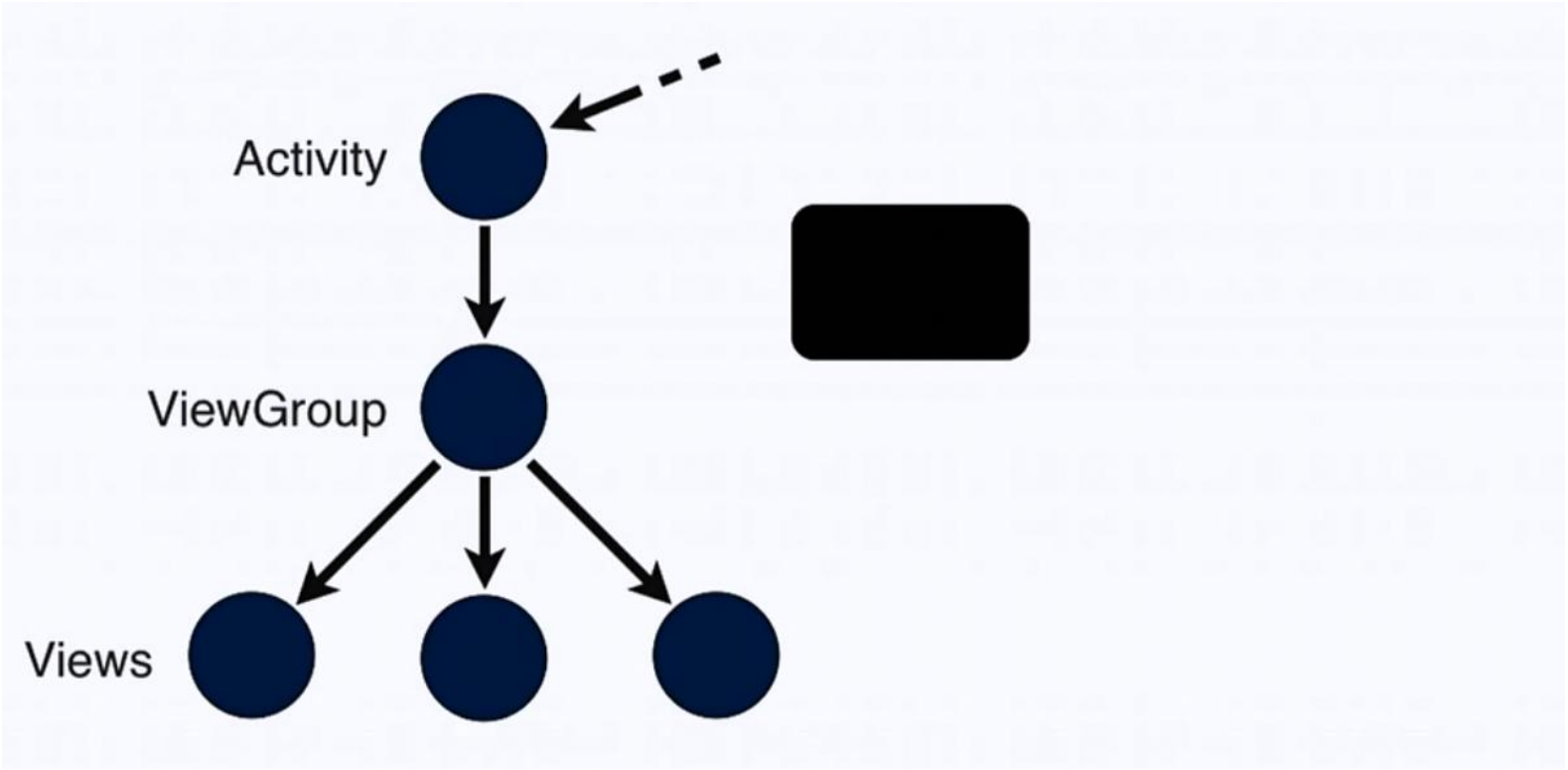
GC and memory leak

- GC don’t eliminate memory leaks



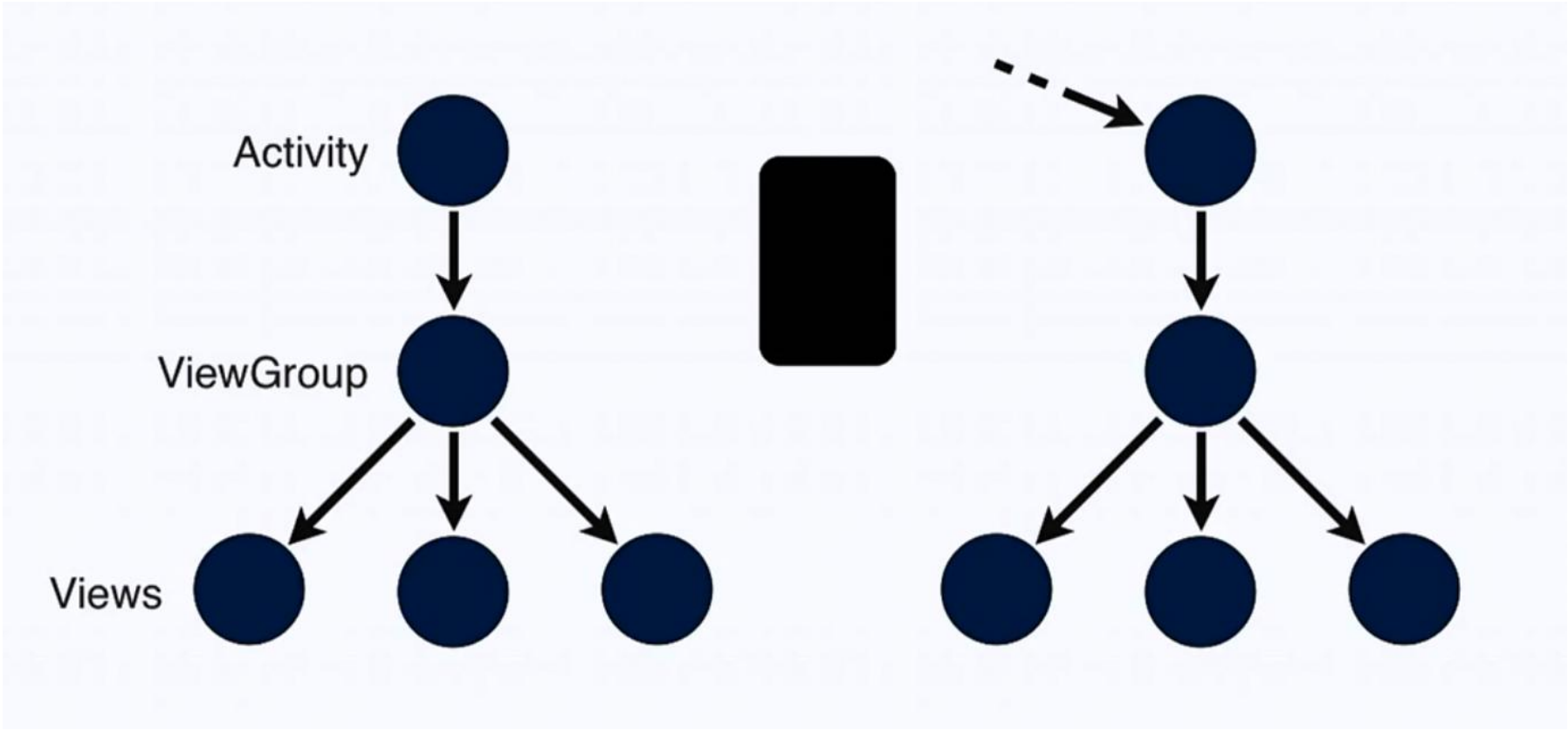
GC and memory leak

- GC don’t eliminate memory leaks



GC and memory leak

- GC don’t eliminate memory leaks



Web resources

- http://handycodeworks.com/wp-content/uploads/2011/02/linux_versus_android.pdf
- http://imsciences.edu.pk/serg/wp-content/uploads/2010/10/1st_Analysis-of-Dalvik-VM.pdf
- http://developer.android.com/guide/basics/what-is-android.html
- http://www.youtube.com/watch?v=v9S5EO7CLjo&feature=plcp&context=C4987524VDvjVQa1PpcFMzwqYlYKVxDu4gnCeJXiKoUpElRjToItM%3D
- http://www.youtube.com/watch?v=FJDP_0Mrb-w&feature=plcp&context=C4fd0520VDvjVQa1PpcFMzwqYlYKVxDnX2uT7xWXvtxdVeqHPNbas%3D
- http://www.youtube.com/watch?v=DTcZPE8Twpg&feature=plcp&context=C491f3faVDvjVQa1PpcFMzwqYlYKVxDrecTG8rMjGpBalFGHi28ro%3D
- http://www.youtube.com/watch?v=YLVbLVtjDDw&feature=plcp&context=C4d0c4bdVDvjVQa1PpcFMzwqYlYKVxDkn3XVQj5fvUQ0TSylfUOXo%3D
- http://www.youtube.com/watch?v=N1aCo5LvMf8&feature=plcp&context=C4faafecVDvjVQa1PpcFMzwqYlYKVxDj0E-nzKHYt7OKR5Fpzm6hM%3D
- http://www.youtube.com/watch?v=cdvaPyq_eBU&feature=plcp&context=C45fa262VDvjVQa1PpcFMzwqYlYKVxDq3n_R6n1AT2qwMmEa7vOVs%3D
- http://www.youtube.com/watch?v=ScW4zSeexvo&feature=plcp&context=C4506f63VDvjVQa1PpcFMzwqYlYKVxDtFmHtj77FF_ysD_ezxpSQ4%3D
- http://www.youtube.com/watch?v=49L7z3rxz4Q&feature=plcp&context=C435e62fVDvjVQa1PpcFMzwqYlYKVxDntPXXpvpxet3Idf8_GhG8E%3D
- http://www.youtube.com/playlist?list=PL586D322B5E2764CF&feature=plcp

