]HackingTeam[

hackingteam.com

PROGRAMMING AND HACKING ANDROID

Fabrizio Cornelli

HT

FABRIZIO CORNELLI

zeno@hackingteam.com

] CV [

Filibusta LUG (1997)

Laurea a Crema (2002)

Direttore tecnico, Enterprise Tech

QA Manager, HT

PRIMA REGOLA DI HT

NON SI PARLA DI HT

]HackingTeam[

Fondata nel 2003

...anche Internet sbaglia

Sommario

Diventare un Ingegnere del Software

Pensare come un Hacker

Come scrivere un APT su Android

INGEGNERE DEL SOFTWARE

"finché va quel tant che basta,
tùchel no, senò el se guasta"

] Come si diventa [

Università

Letteratura

Comunità

Progetti open source

Alcuni dicono

Scegline due: fatto bene, veloce, che costa poco

Chiunque può scrivere codice che un computer riesce a capire. Un buon programmatore riesce a scrivere un codice comprensibile anche per altri programmatori. [M. Fowler]

HACKER

“Ogni tecnologia sufficientemente avanzata è indistinguibile dalla magia”

] Come si diventa [

Sicurezza, architetture e crittografia

H / V / C forum

Conferenze (def con, blackhat, hack-tic)

Ambienti protetti di test

Alcuni dicono

Una persona intelligente risolve i problemi, una saggia li evita.

] Ing. SW | Hacker [

Costruttivo | Decostruttivo

]HackingTeam[

] Valori condivisi [

Programmazione

Studio intenso e quotidiano

Attenzione ai dettagli

Fantasia e immaginazione

] Ipotesi antropologica [

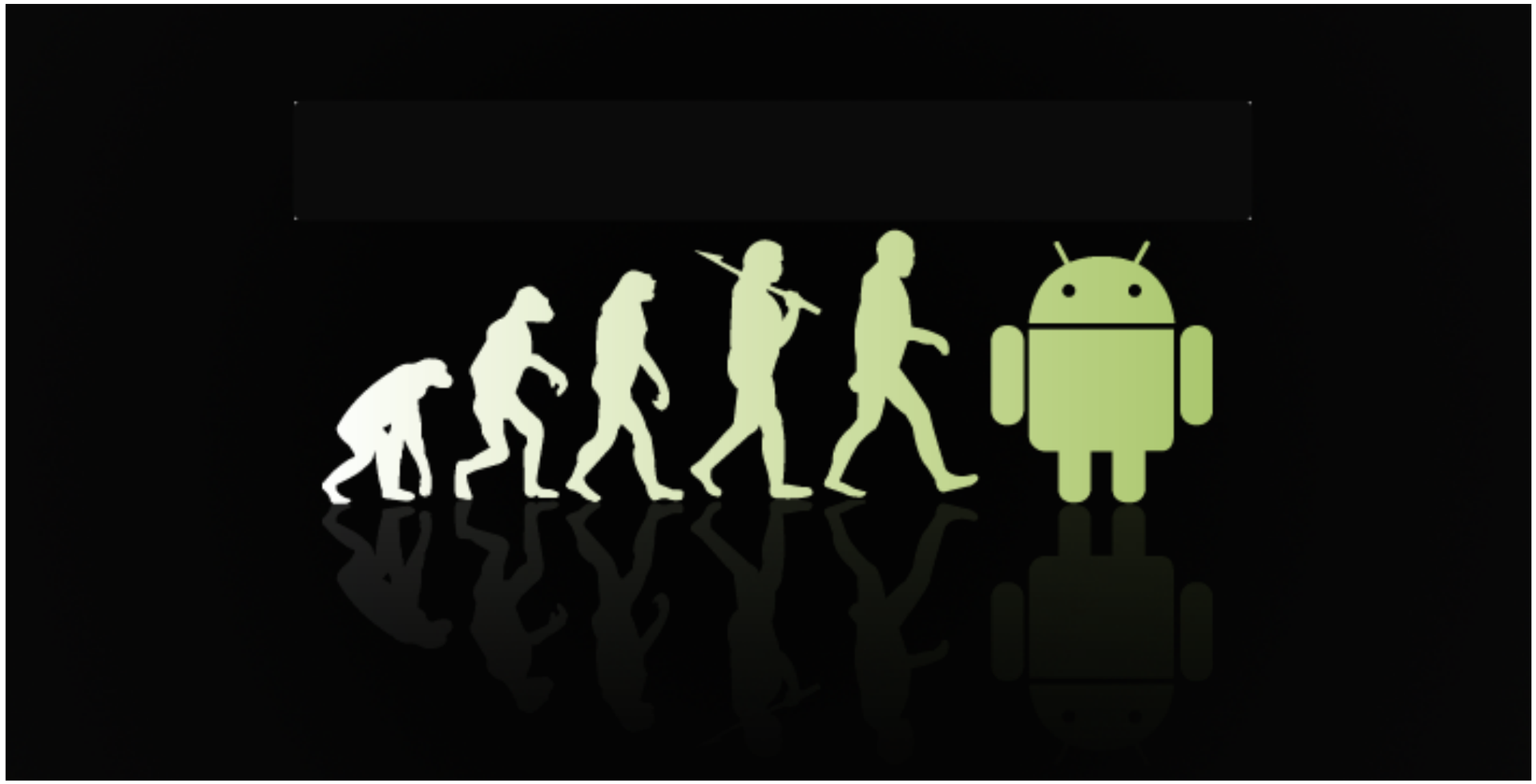
] Allevatore | Cacciatore [

Stanziale	Opportunista
Crede nella Tradizione	Innovativo
Paziente	Dotato di Iniziativa
Collaborativo	Indipendente

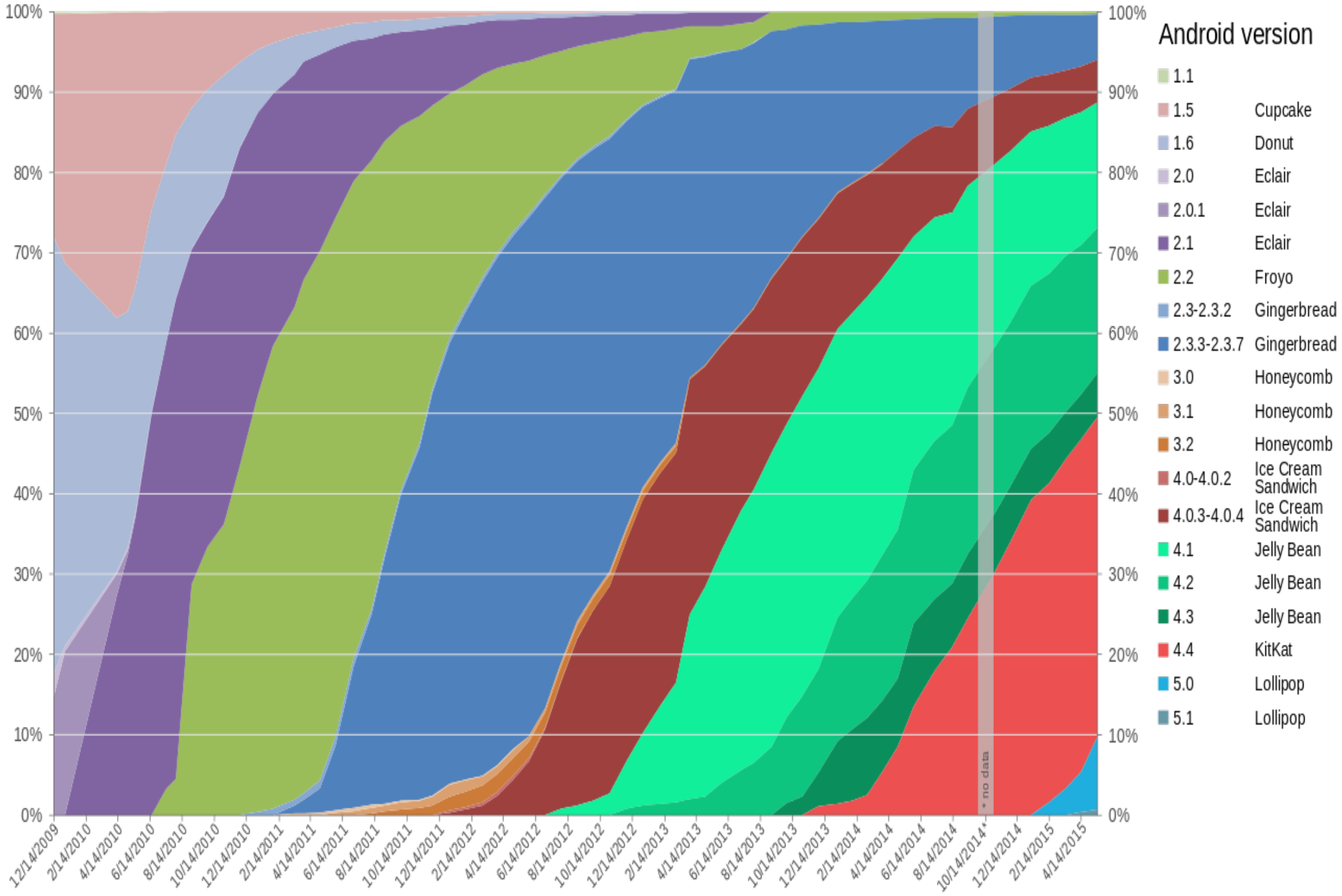
] Programmatore [

Pensa come un **hacker**: costruisci un prototipo innovativo

Scrivi come un **ingegnere del software**: fai evolvere l'idea in un prodotto



ARCHITETTURA ANDROID


100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
12/14/2009
2/14/2010
4/14/2010
6/14/2010
8/14/2010
10/14/2010
12/14/2010
2/14/2011
4/14/2011
6/14/2011
8/14/2011
10/14/2011
12/14/2011
2/14/2012
4/14/2012
6/14/2012
8/14/2012
10/14/2012
12/14/2012
2/14/2013
4/14/2013
6/14/2013
8/14/2013
10/14/2013
12/14/2013
2/14/2014
4/14/2014
6/14/2014
8/14/2014
10/14/2014*
12/14/2014
2/14/2015
4/14/2015
* no data
Android version
1.1
1.5 Cupcake
1.6 Donut
2.0 Eclair
2.0.1 Eclair
2.1 Eclair
2.2 Froyo
2.3-2.3.2 Gingerbread
2.3.3-2.3.7 Gingerbread
3.0 Honeycomb
3.1 Honeycomb
3.2 Honeycomb
4.0-4.0.2 Ice Cream Sandwich
4.0.3-4.0.4 Ice Cream Sandwich
4.1 Jelly Bean
4.2 Jelly Bean
4.3 Jelly Bean
4.4 KitKat
5.0 Lollipop
5.1 Lollipop

Applications Framework

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

Libraries

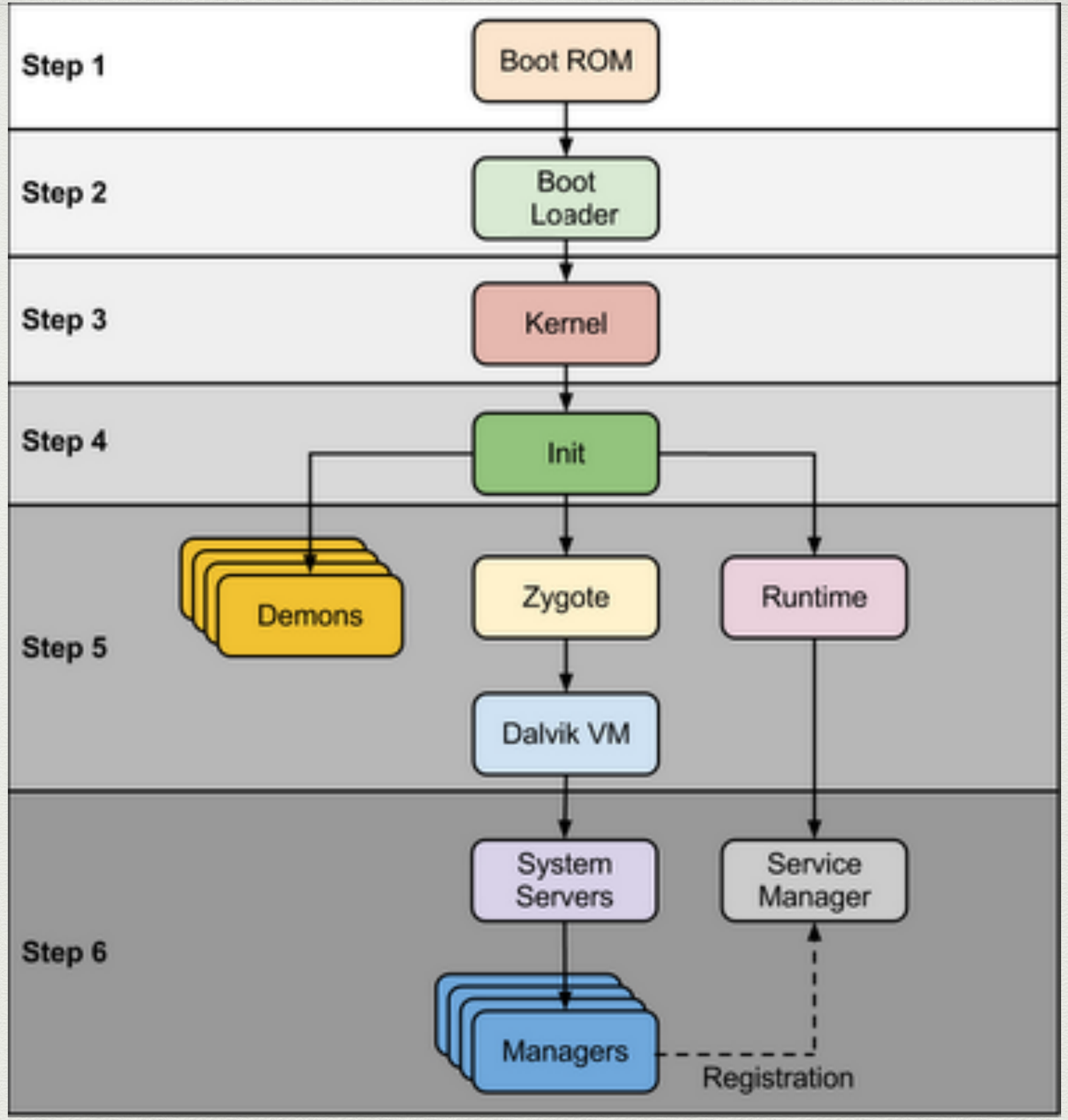
- Surface Manager
- Media Framework
- SQLite
- OpenGL | ES
- FreeType
- WebKit
- SGL
- SSL
- Libc

Android Runtime

- Core Libraries
- Dalvik Virtual Machine

Linux Kernel

- Display Driver
- Camera Driver
- Flash Memory Driver
- Binder (IPC) Driver
- Keypad Driver
- WiFi Driver
- Audio Drivers
- Power Management


Step 1
Boot ROM
Step 2
Boot Loader
Step 3
Kernel
Step 4
Init
Step 5
Demons
Zygote
Runtime
Dalvik VM
Step 6
System Servers
Service Manager
Managers
Registration

Boot

ROM / Boot loader

kernel

drivers

fs

init (user space)

zygote / dalvik /ART

daemons

GC

Dalvik

Process VM

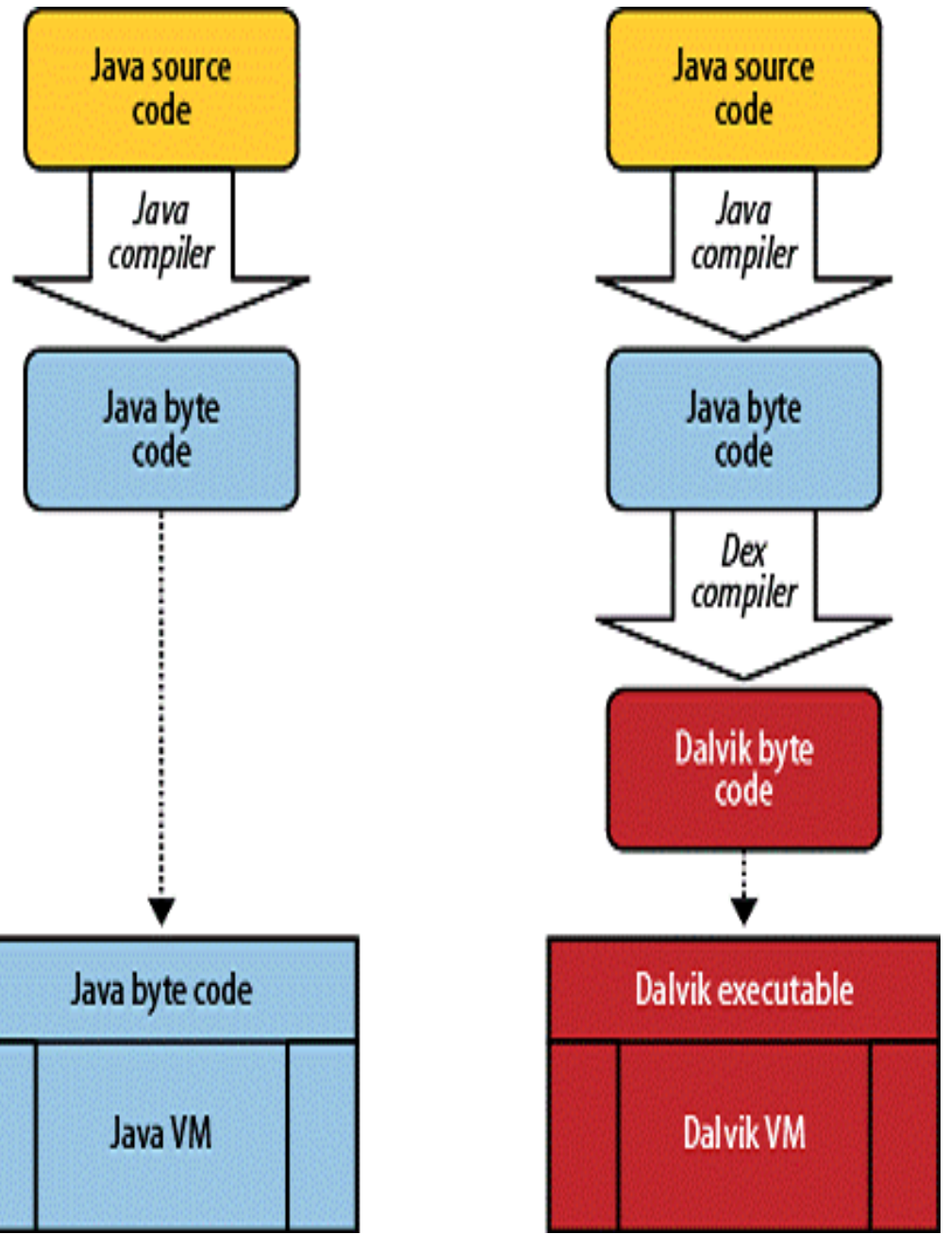
register based

Java -> bytecode -> dex

APK

Each app its copy

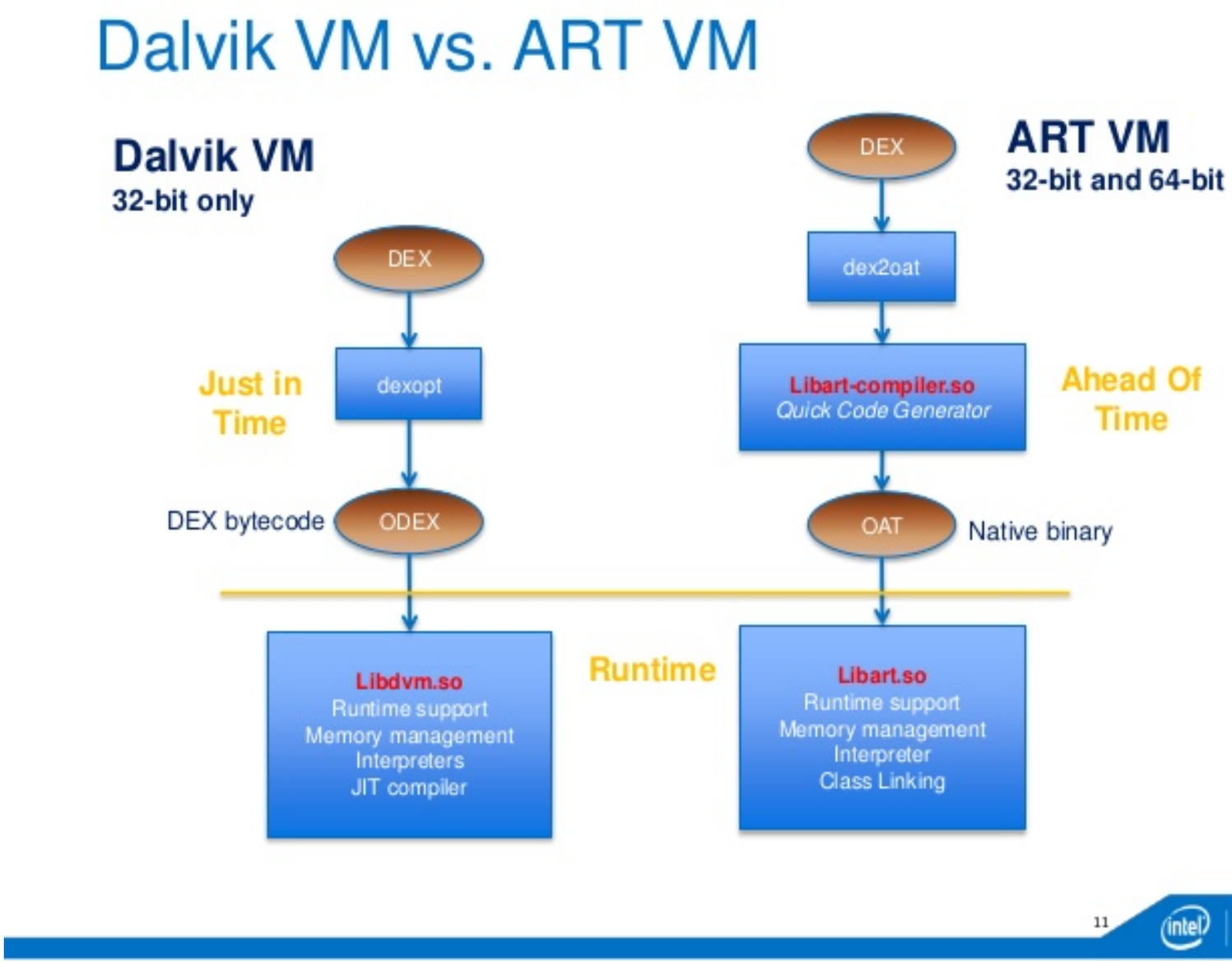
JIT



] Android Runtime (ART) [

.dex -> elf

AOT



APK

Zip file

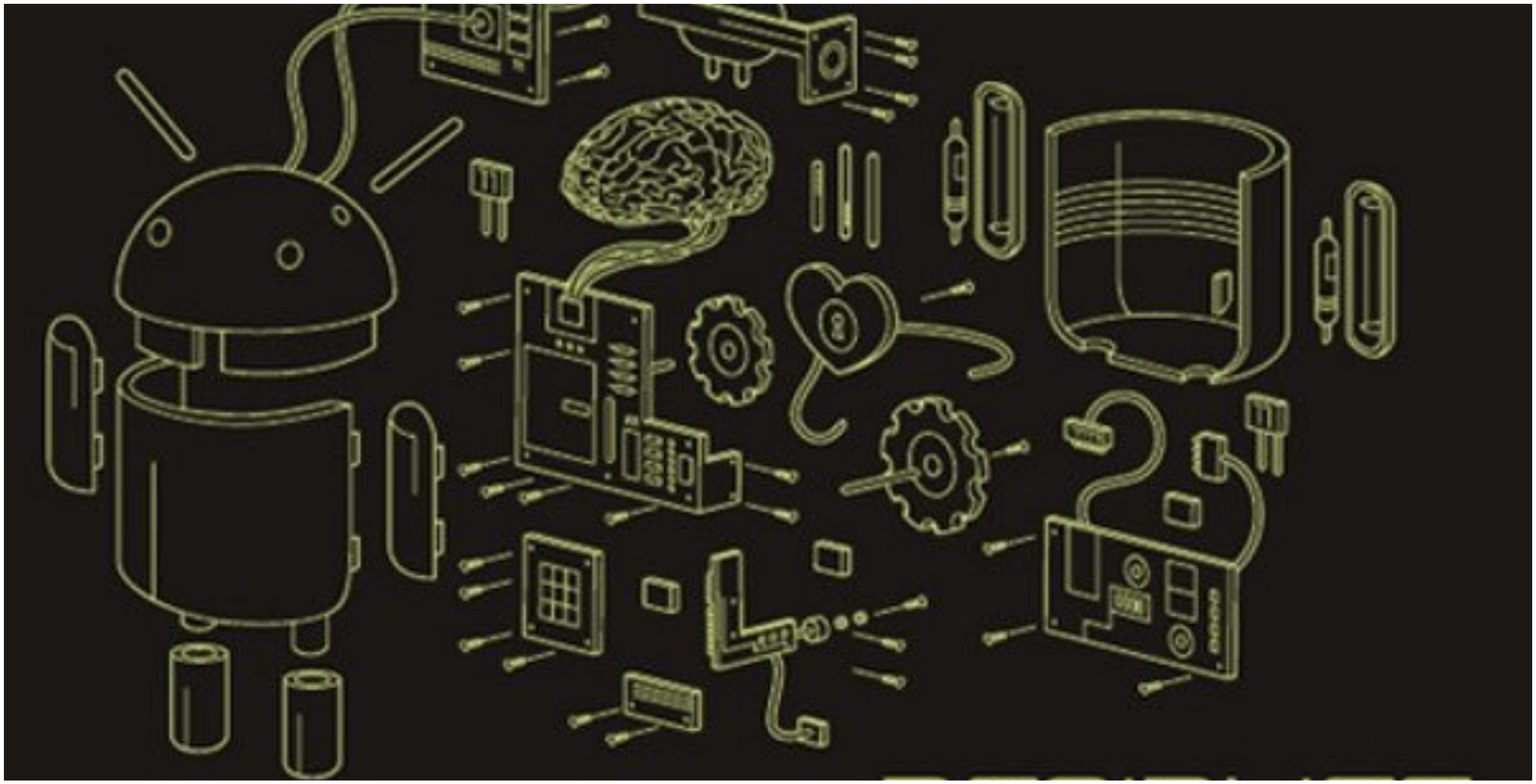
classes.dex : Code, *.class

Manifest

Resources / Assets /Licence

Libs

Signature



ANDROID DEVELOPMENT

] Android Studio [

Piattaforma IntelliJ

gradle / ant

emulatori

ADB

Android Debug Bridge

Richiede che il device sia in "Debug Mode"

- shell
- pull
- push
- install
- logcat
- reboot

REVERSING

Strumenti

Decompilatori : jd-gui, dad, jeb

Dissettori Apk : androguard, apktool

Piattaforme di reversing: IDA, Radare

Analizzatori di rete: Wireshark, ettercap, tcpdump, burp

] apktool [

decode apk

build apk

Internamente usa *smali / baksmali*

Serve usare *jarsigner*

] smali [

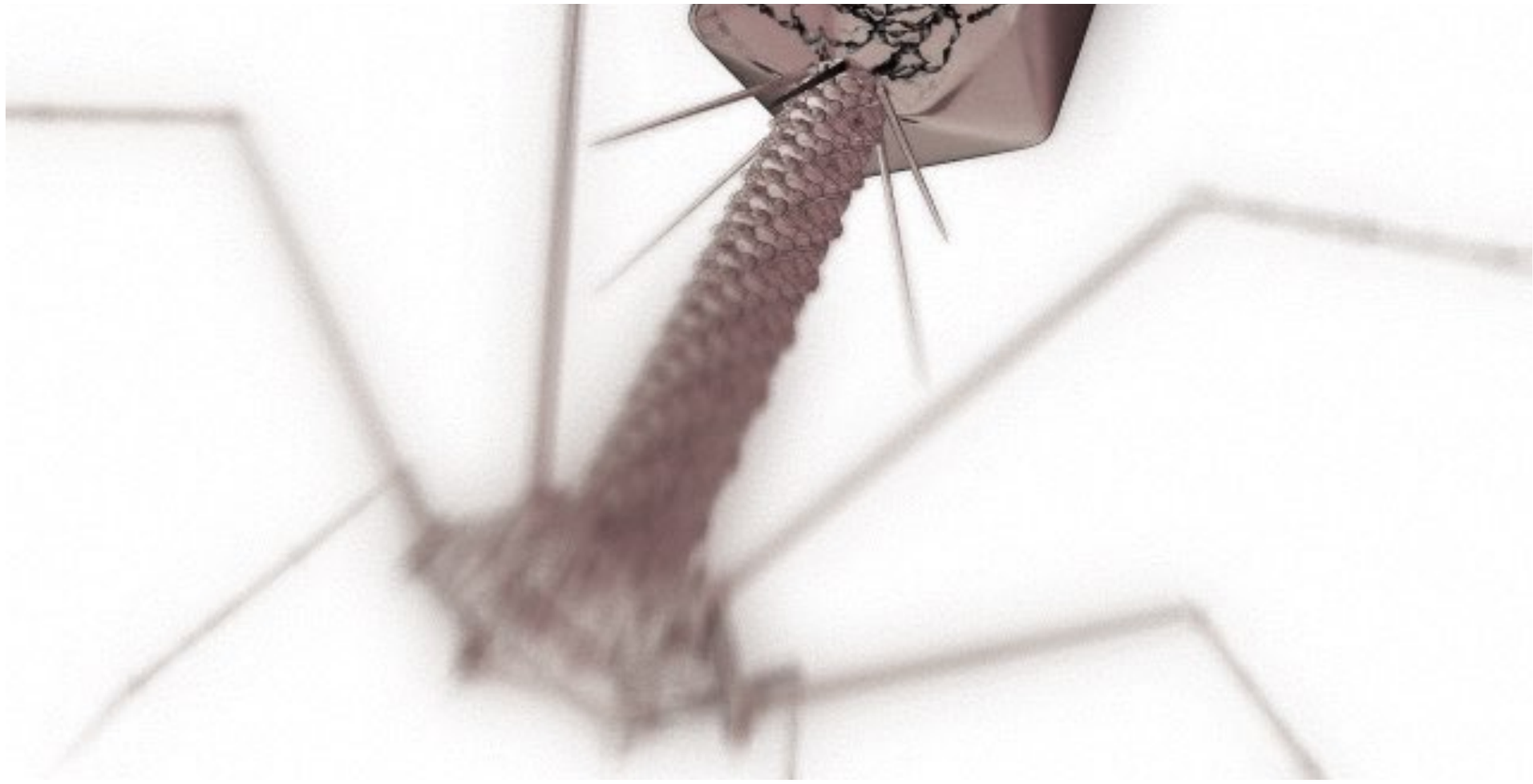
```
.method public static main([Ljava/lang/String;)V
    .registers 2

    sget-object v0, Ljava/lang/System;->out:Ljava/io/PrintStream;

    const-string   v1, "Hello World!"

    invoke-virtual {v0, v1}, Ljava/io/PrintStream;->println(Ljava/lang/String;)V

    return-void
.end method
```



SCRIVERE UN APT

Advanced Persistence Threats

] Anatomia [

Software

Target specifico

Nascosto

Raccoglie dati nel tempo

Command and Control

Ciclo di vita

Configurazione

Costruzione del pacchetto

Installazione

Esecuzione

Persistenza

Raccolta e inoltro dei dati

Disinstallazione

] Imparare dal malware [

http://contagiominidump.blogspot.it/

Virus Total

Androguard DatabaseAndroidMalwares

HACKING ANDROID

] Ottenere i privilegi [

Privilegi dichiarati nel Manifest

Flash dell'OS

Usare un *local to root* exploit

Usare un *system* exploit

] Esecuzione al boot [

Manifest

OS (root)

] Raccogliere i dati [

Android API

Prendere le informazioni sul fs (root)

Cercare le informazioni in memoria (root)

] Comunicazione [

Canale di comunicazione protetto e nascosto

Android API?

Strumenti

Configurazione:

Costruzione del pacchetto:

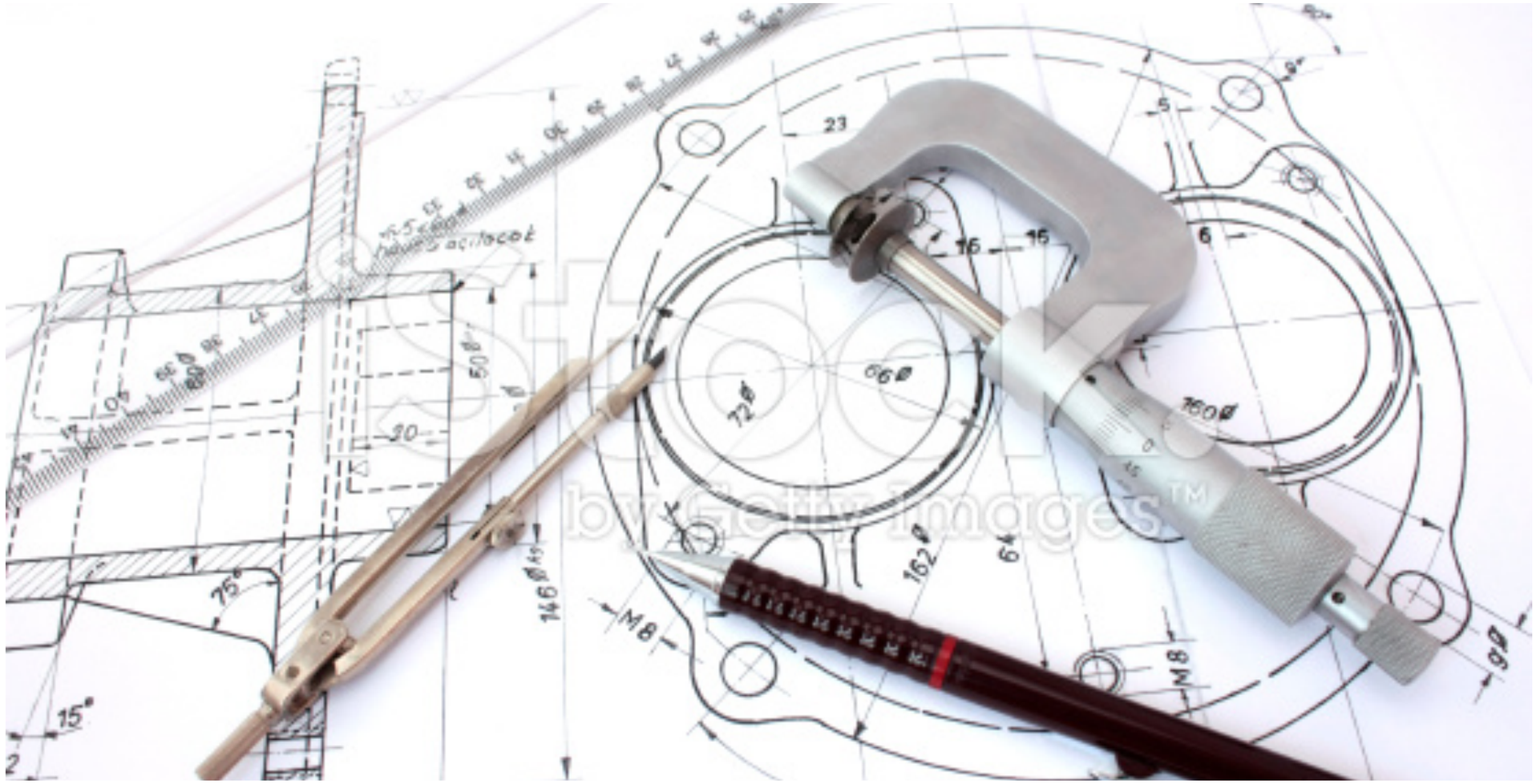
Installazione:

Esecuzione:

Persistenza:

Raccolta e inoltro dei dati:

Disinstallazione:



REALIZZARE UN PRODOTTO

Agile programming

] Fasi dello sviluppo [

	Software Engineer	Hacker
Progettazione	++	++
Prototipizzazione	+	+++
Architettura	+++	+
Test e Mantenimento	+++	+

] Mantenere il codice [

Versionamento

Continuous integration

Automatic build

Test automatici e code coverage

Installazione automatica

] Protezione dagli hacker [

Anti reversing

Polimorfismo

Cifratura e offuscamento

Packing

Virtualizzazione del codice

]HackingTeam[

NOW HIRING

you didn’t see anything...

Links

http://learnlinuxconcepts.blogspot.com/2014/02/android-boot-sequence.html

http://www.amazon.com/Android-Hackers-Handbook-Joshua-Drake/dp/111860864X

http://www.nyxbone.com/malware/android_tools.html

Stackoverflow

xdadevelopers