

Conceptes Avançats de Sistemes Operatius

Facultat d'Informàtica de Barcelona
Dept. d'Arquitectura de Computadors

Curs 2013/14 Q1

Dispositius mòbils



Departament d'Arquitectura de Computadors

FIB

Índex

- Dispositius mòbils
- Característiques del SO
- Android

Dispositius mòbils

- Mercat creixent de dispositius "de butxaca"
 - Telèfons
 - Tablets
 - PDAs
- Necessitat de tenir un software cada cop més semblant a un desktop
- Comunicacions
 - Telefonía, Wifi, Bluetooth...
- Nous dispositius
 - Touchpad, GPS, acceleròmetres...

Varietat de marques i models



Substancial varietat de sistemes

- Apple iOS
- Google Android (many companies)
- Microsoft Windows (Nokia WindowsPhone)
- NokiaOS (Nokia), Symbian (Accenture)
- RIM BlackBerry (QNX)
- Samsung Bada (2009, Linux)
- Firefox OS (Mozilla, based on Linux and Android (?))

Substancial varietat de sistemes

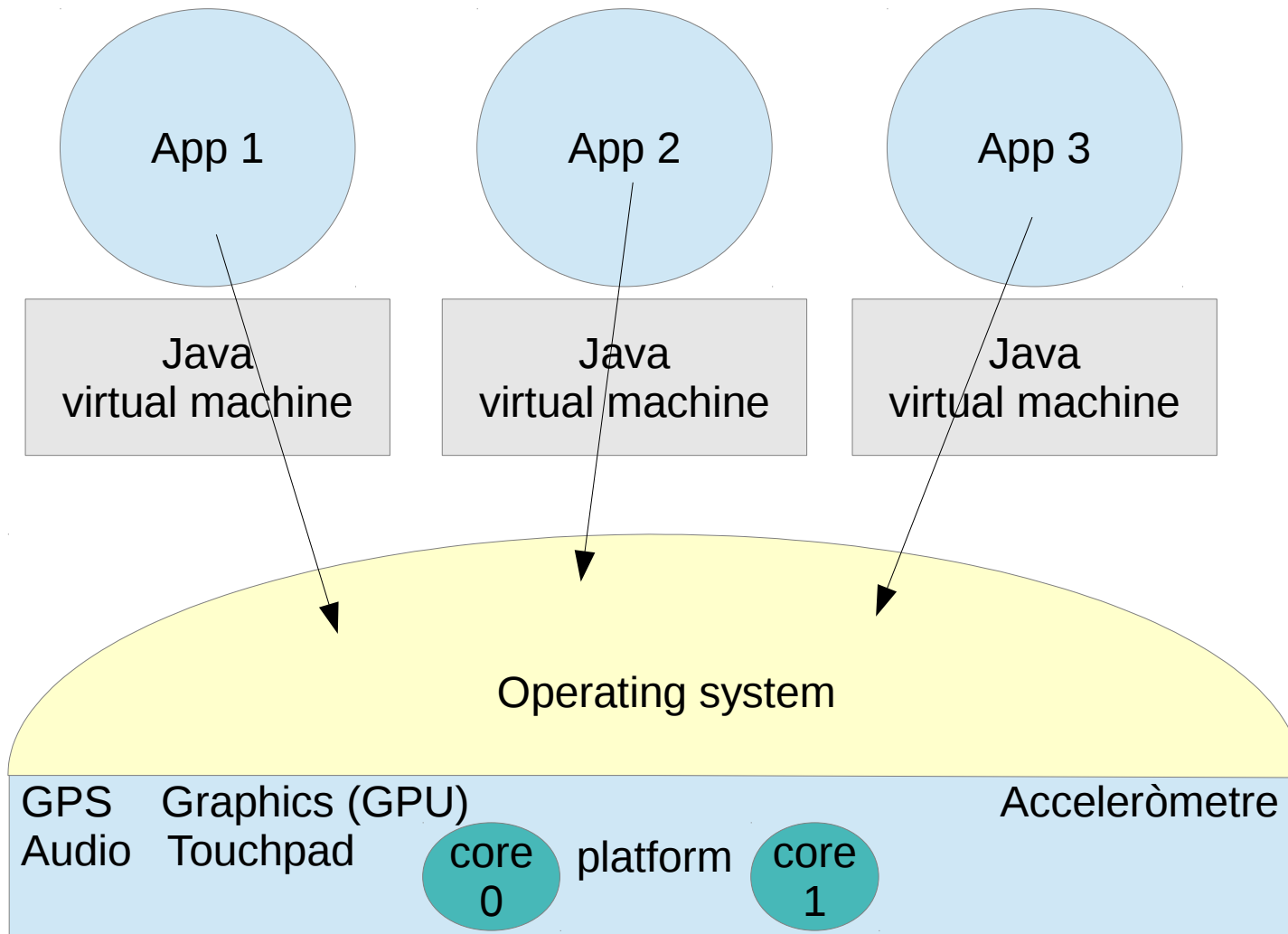
- Tornada enrere
 - Entorns d'una sola aplicació
 - Multitasca cooperativa
 - Sistemes tancats
 - Especialment pel que fa als controladors de dispositiu
 - Android és lliure, però els controladors de GPS, touchpad, wifi, ... són tancats
 - SMP per necessitat
 - Suport per acceleradors
- Alguns basats en Linux (simplificat?) modificat
 - Android, Bada, Firefox OS

Multitasca cooperativa

- Quan un procés no té res a fer...
 - o pot esperar un temps per realitzar la següent tasca
 - usa una crida de la interfície per deixar el processador
 - pause, wait, **yield**, nop...
- No preemptiva
 - Si el procés no fa yield, el SO no passa el processador a cap altre procés
- Interrumpible
 - Assegurant-se que els processos es poden matar
 - Tecla vermella en els smartphones

Entorn mòbil

- Molts entorns estan basats en Java



Processadors embedded

- IBM PowerPC
 - PPC 440 / 450 / Power A2 => BG/L/P/Q
 - 32 => 64 bits
 - UNIX/Linux
- MIPS
 - R3000/R4000... R16000/MIPS32/64
 - 32 => 64 bits
 - UNIX/Linux
- ARM v1 => v8, produït per molts fabricants
 - v7, VFP, SIMD Neon, Jazelle, Thumb
 - v8 (2011, 64-bits)
 - +MALI GPU, Qualcomm Adreno, Imagination PowerVR SGX...
 - Windows8/UNIX/Linux

Processadors embedded

- Intel embedded
 - Atom Zxxxx (x86, x86_64), alguns Xeon
 - ...
- AMD
 - Geode, G-Series, R-series, alguns Opteron
- Texas Instruments
 - ARM OMAP
 - DSP (Digital Signal processors)

Processadors embedded

- Xylinx microblaze
 - per FPGAs
- Altera nios2
 - FPGAs

Android

- Desenvolupat per Google i Open Handset Alliance
 - 2007 – actualitat
 - Android 4.4
- Implementat per les arquitectures ARM, MIPS, x86(_64)
- Inclou canvis al kernel de Linux
 - Actualment 3.2
- Hi ha la intenció d'incorporar lentament els canvis a la branca principal de Linux?
 - De fet, amb moltes discussions i sense massa èxit

http://elinux.org/Android_Kernel_Features

http://developer.android.com/about/versions/kitkat.html

Android 4.4 kit-kat

- SELinux - Security Enhanced – Enforcing mode

http://selinuxproject.org/page/Developers

SELinux development is primarily done on the [NSA SELinux mailing list](#) (Unofficial searchable list archive) !!!



The screenshot shows the NSA Central Security Service website. The header includes the NSA and CSS logos and the slogan "Defending Our Nation. Securing The Future." The navigation menu includes: HOME, ABOUT NSA, ACADEMIA, BUSINESS, CAREERS, INFORMATION ASSURANCE, RESEARCH, PUBLIC INFORMATION, and COMMITMENT. The left sidebar lists various research topics, with "Security Enhanced Linux" highlighted. The main content area shows the breadcrumb "Home > Research > Selinux" and a search box. The title is "Security-Enhanced Linux". The text describes the NSA's involvement in computer security research, mentioning the Trusted Systems Research Group and the Flask architecture.

NATIONAL SECURITY AGENCY   CENTRAL SECURITY SERVICE

Defending Our Nation. Securing The Future.

HOME ABOUT NSA ACADEMIA BUSINESS CAREERS INFORMATION ASSURANCE **RESEARCH** PUBLIC INFORMATION COMMITMENT

Research

- The Next Wave (TNW)
- ▶ **Security Enhanced Linux**
- What's New
- Frequently Asked Questions
- Background
- Documents
- License
- Download
- Participating
- Mail List
- Remaining Work
- Contributors
- Related Work
- Press Releases
- Information Assurance Research
- Mathematical Sciences Program

Home > Research > Selinux

Security-Enhanced Linux

As part of its [Information Assurance](#) mission, the National Security Agency has long been involved with the computer security research community in investigating a wide range of computer security topics including operating system security. Recognizing the critical role of operating system security mechanisms in supporting security at higher levels, researchers from NSA's [Trusted Systems Research Group](#), formerly the National Information Assurance Research Laboratory, have been investigating an architecture that can provide the necessary security functionality in a manner that can meet the security needs of a wide range of computing environments.

End systems must be able to enforce the separation of information based on confidentiality and integrity requirements to provide system security. Operating system security mechanisms are the foundation for ensuring such separation. Unfortunately, existing mainstream operating systems lack the critical security feature required for enforcing separation: mandatory access control. As a consequence, application security mechanisms are vulnerable to tampering and bypass, and malicious or flawed applications can easily cause failures in system security.

The results of several previous [research projects](#) in this area have yielded a strong, flexible mandatory access control architecture called [Flask](#). A reference implementation of this architecture was first integrated into a

Android 4.4 kit-kat

- Millora les eines d'anàlisi del rendiment
 - procstats
 - meminfo
- Millora l'accés a impressores
- Hardware sensor batching
- VPN per usuari (enloc de global)
 - per entorns multi-usuari

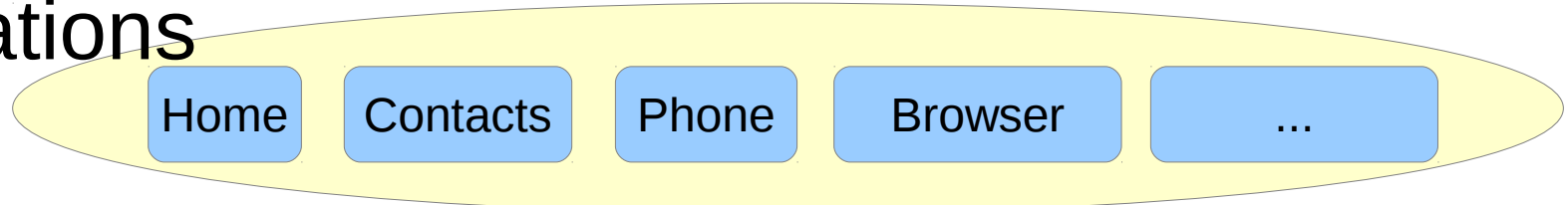
```
Home: 0.00%
(Cached): 67% (54MB-56MB-67MB/44MB-47MB-62MB over 32)
* com.google.android.gms / u0a7:
  TOTAL: 28% (10MB-11MB-11MB/8.0MB-8.1MB-8.2MB over 16)
  Top: 27% (10MB-11MB-11MB/8.0MB-8.1MB-8.2MB over 16)
  Imp Fg: 1.2%
  Service: 0.14%
  Receiver: 0.01%
  Home: 0.00%
(Last Act): 0.00%
(Cached): 70% (10MB-11MB-11MB/8.0MB-8.1MB-8.2MB over 15)
* com.google.android.apps.books / u0a26:
  TOTAL: 22% (50MB-80MB-113MB/43MB-63MB-82MB over 12)
  Top: 22% (50MB-80MB-113MB/43MB-63MB-82MB over 12)
  Service: 0.00%
(Last Act): 4.6% (85MB-85MB-85MB/77MB-77MB-77MB over 2)
(Cached): 58% (78MB-81MB-82MB/70MB-73MB-74MB over 4)
* com.android.settings / 1000:
  TOTAL: 2.8% (14MB-15MB-16MB/11MB-12MB-13MB over 3)
  Top: 2.8% (14MB-15MB-16MB/11MB-12MB-13MB over 3)

261 kB: bridgemgrd (pid 183)
236 kB: dhcpcd (pid 13422)
232 kB: adbd (pid 4572)
227 kB: debuggerd (pid 172)
199 kB: rmt_storage (pid 170)
194 kB: qseecomd (pid 252)
188 kB: mpdecision (pid 1466)
183 kB: installld (pid 178)
140 kB: healthd (pid 167)
114 kB: servicemanager (pid 168)
103 kB: qseecomd (pid 193)
94 kB: logwrapper (pid 990)

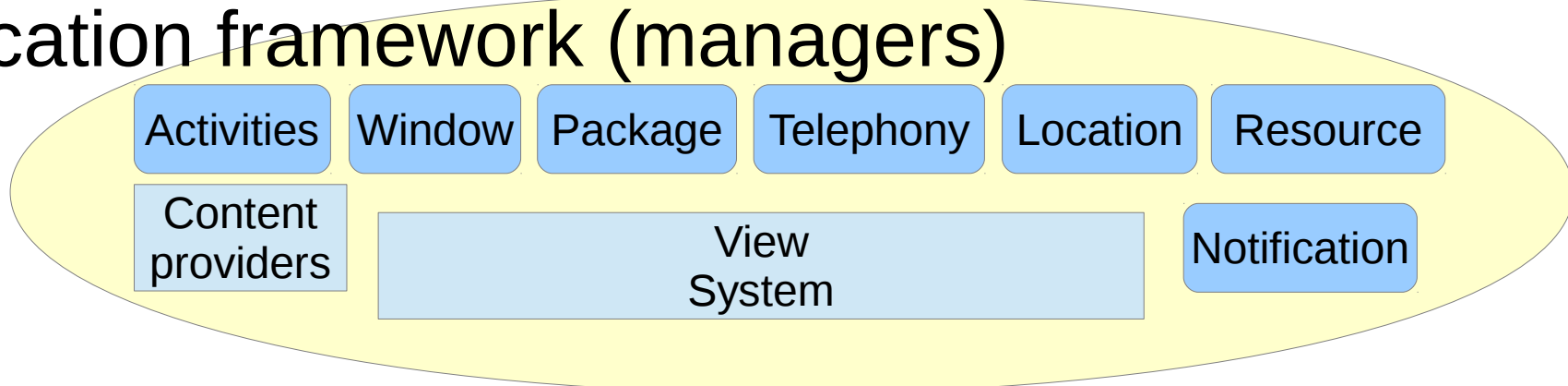
Total PSS by OOM adjustment:
43529 kB: Native
12363 kB: mediaserver (pid 177)
5889 kB: mm-qcamera-daemon (pid 195)
5480 kB: sensors.qcom (pid 187)
3842 kB: surfaceflinger (pid 174)
3488 kB: zygote (pid 175)
```

Com és Android?

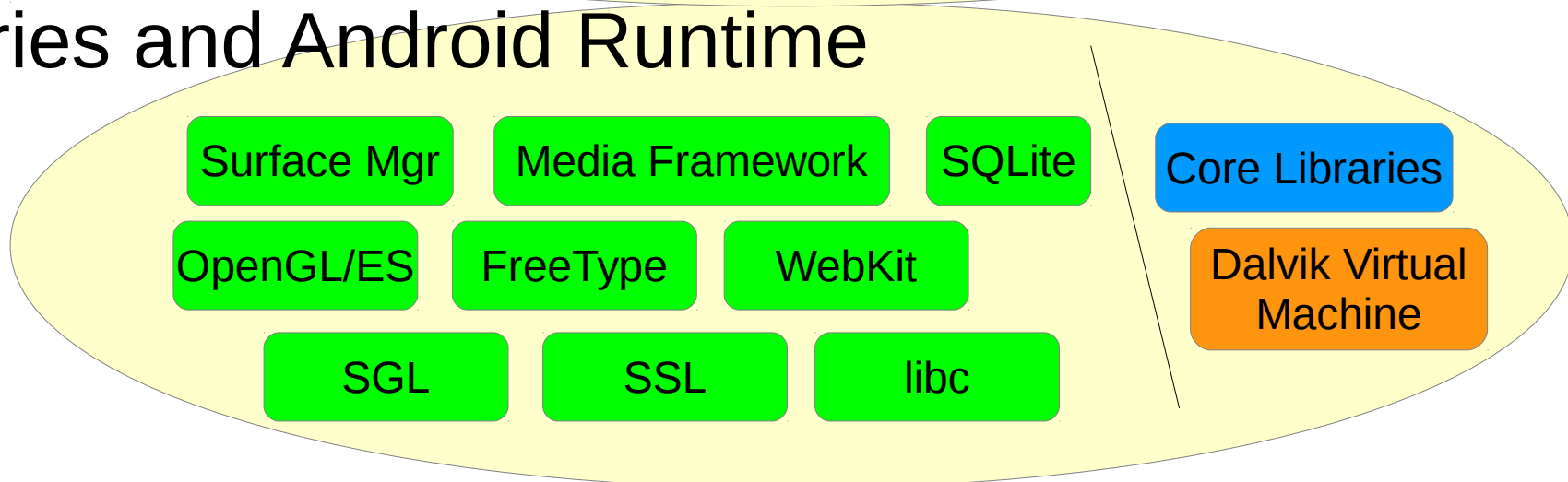
- Applications



- Application framework (managers)



- Libraries and Android Runtime



- Linux



Aplicacions

- Compilades en fitxers comprimits (.apk)
 - = format zip
- Cada aplicació està tancada en un *security sandbox*, principi de *least privilege*
 - UID diferent, cada aplicació pot accedir només als seus fitxers
 - Màquina virtual diferent
 - Procés Linux propi
 - És destruïda quan falta memòria

Aplicacions

- Cada aplicació
 - Té la seva pròpia MV
 - Córrer en el seu propi usuari (UID propi)
 - Protecció entre aplicacions
 - Dificulta compartir dades entre aplicacions
- És possible
 - Que dues aplicacions comparteixin UID
 - Que dues aplicacions comparteixin procés i MV
 - Han d'estar signades amb el mateix certificat
- El permís per accedir als recursos l'ha de donar l'usuari
 - A l'instal·lar l'aplicació
 - En el moment de l'accés (*pop-up?*, només en altres sistemes?)

<http://developer.android.com/guide/components/fundamentals.html>

Aplicacions

- Components
 - Activities, cada pantalla amb la seva UI (user interf.)
 - Les activitats poden engegar-se des d'altres aplicacions
 - e.g. l'aplicació de la càmera pot engegar l'activitat d'enviar un missatge
 - Services, activitat en background (sense pantalla)
 - Implementen operacions llargues en temps
 - Com a serveis per no bloquejar la UI
 - Donen servei a altres aplicacions

Aplicacions

- Components (cont.)
 - Content providers, per accedir a dades
 - Compartides amb altres aplicacions o no
 - Broadcast receivers, per atendre a events del sistema
 - La bateria està baixa
 - La pantalla s'ha apagat
 - ...

Aplicacions

- Invocació de components
 - e.g. invocar el component de l'aplicació de càmera per fer una foto
 - El sistema engega el procés per a l'aplicació en particular (càmera)
 - Si no és que ja està engegat
 - Instancia les classes que el component necessita
 - Això vol dir que no hi ha funció "main"
 - A través d'Intent Objects
 - Els components estan declarats en cada aplicació:
 - AndroidManifest.xml

Aplicacions

- Activities, services & broadcast receivers són invocats per Intent Objects
 - Component name - qui
 - Class name: "com.example.project.app.YourActivity"
 - Package name: "com.example.project"
 - Action - què
 - CALL, EDIT, SYNC, BATTERY_LOW, HEADSET_PLUG
 - Data - dades sobre les que actuar
 - Document, audio, vídeo, número de telèfon...
 - Categoria, informació addicional i flags

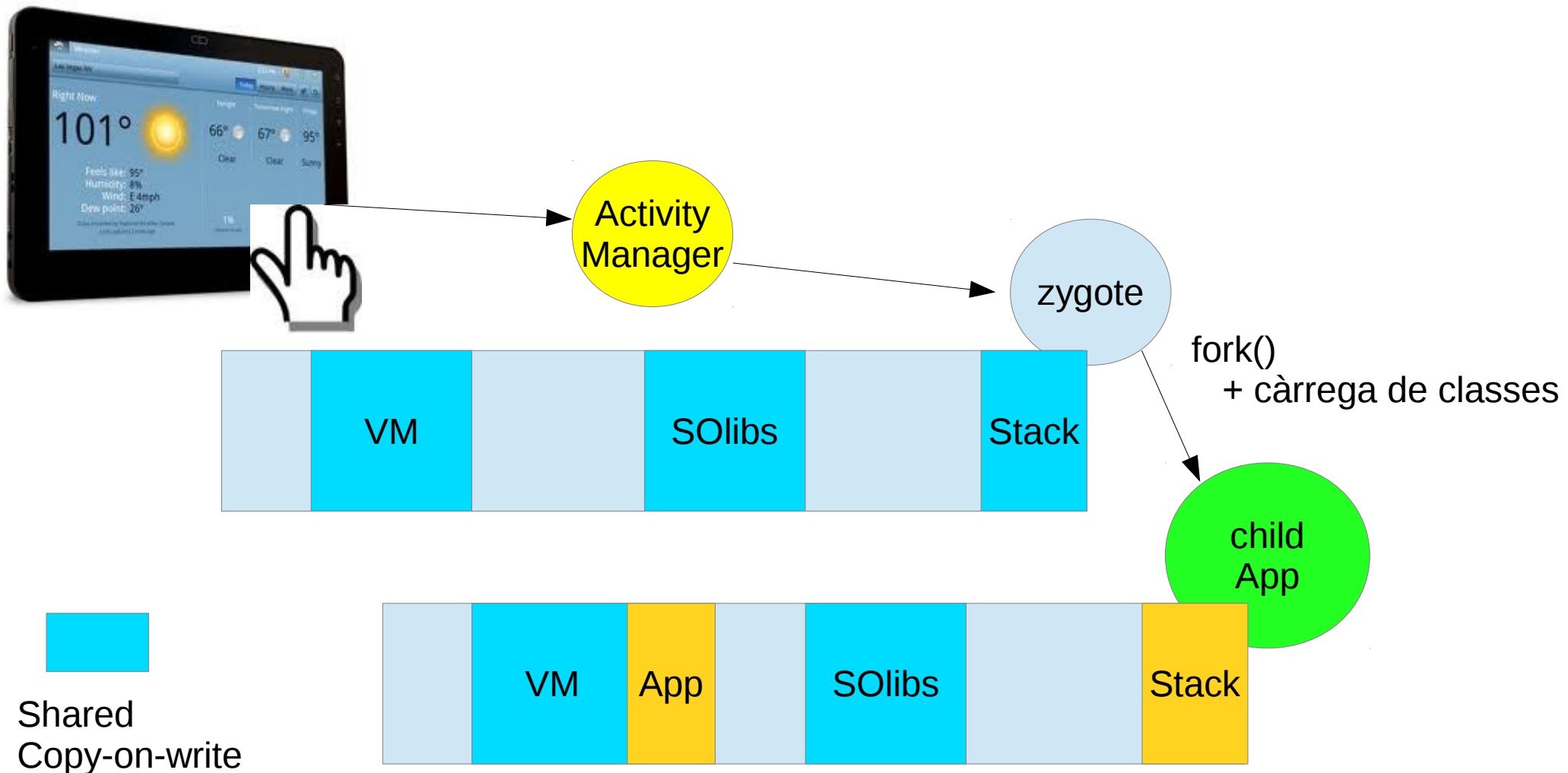
Gestió de processos

- Helper a nivell d'usuari
 - Zygote process: "la cel·lula inicial d'un nou individu"
 - Procés pre-carregat amb la Dalvik VM
 - Càrrega de llibreries dinàmiques ja realitzada
 - Engegar una aplicació:
 - Fer un fork de "zygote" i carregar l'aplicació Java
 - Bona part del procés és compartit
 - Màquina virtual
 - Llibreries compartides

<http://coltf.blogspot.com.es/p/android-os-processes-and-zygote.html>

Gestió de processos

- Com s'engega una aplicació



Gestió de processos

- Gestionats com un *activity stack*
 - Al començar una nova activitat es situa al capdamunt
 - i rep el processador (*resumed*)
 - L'activitat anterior queda en suspens, fins que la nova acaba
 - es para (*stopped*)

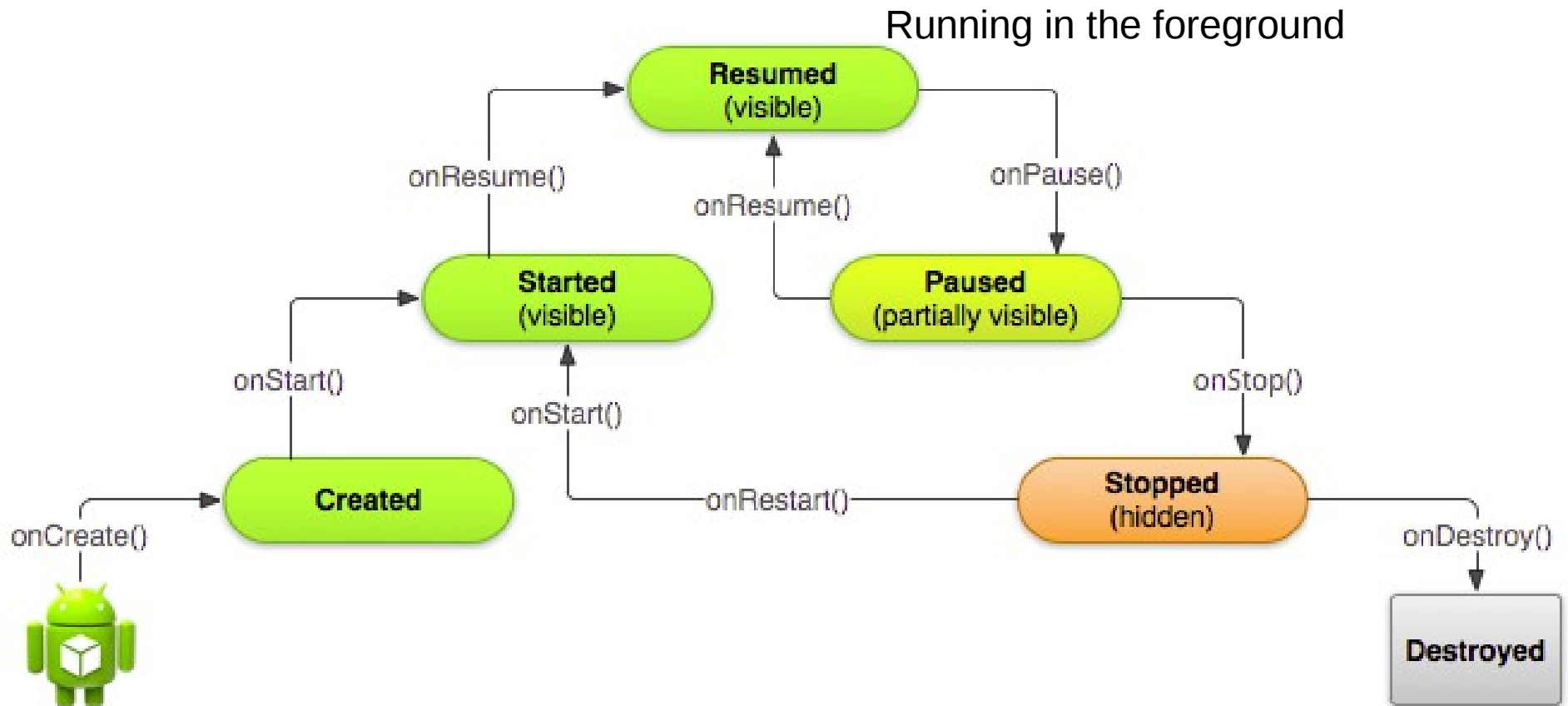
<http://developer.android.com/training/basics/activity-lifecycle/index.html>

Gestió de processos

- Les *activities* passen per 3 estats
 - *Resumed* o en *foreground*
 - *Paused*, si queda parcialment tapada per una altra *activity*
 - Continua viva
 - Pot ser eliminada en cas de falta **extrema** de memòria
 - *Stopped*, si queda totalment tapada per una altra
 - Pot ser continuada si torna a veure's
 - Pot ser eliminada si falta memòria per a realitzar alguna altra activitat

Gestió de processos

- Graf d'estats



"The activity can also return to the resumed state from the Paused and Stopped state."

onRestoreInstanceState()

Gestió de processos

- onCreate()... onStart()... onResume() callbacks
 - Habitualment es reben seguits a l'arrancar l'app
- onPause()..... onResume()
 - Guardar dades si l'usuari espera un auto-save
 - formularis a mig omplir...
 - Restaurar
- onPause()... onStop().....
..... onRestart()... onStart() ...onResume()
 - Guardar dades i alliberar recursos
- onSaveInstanceState() ... onDestroy()

Gestió de processos

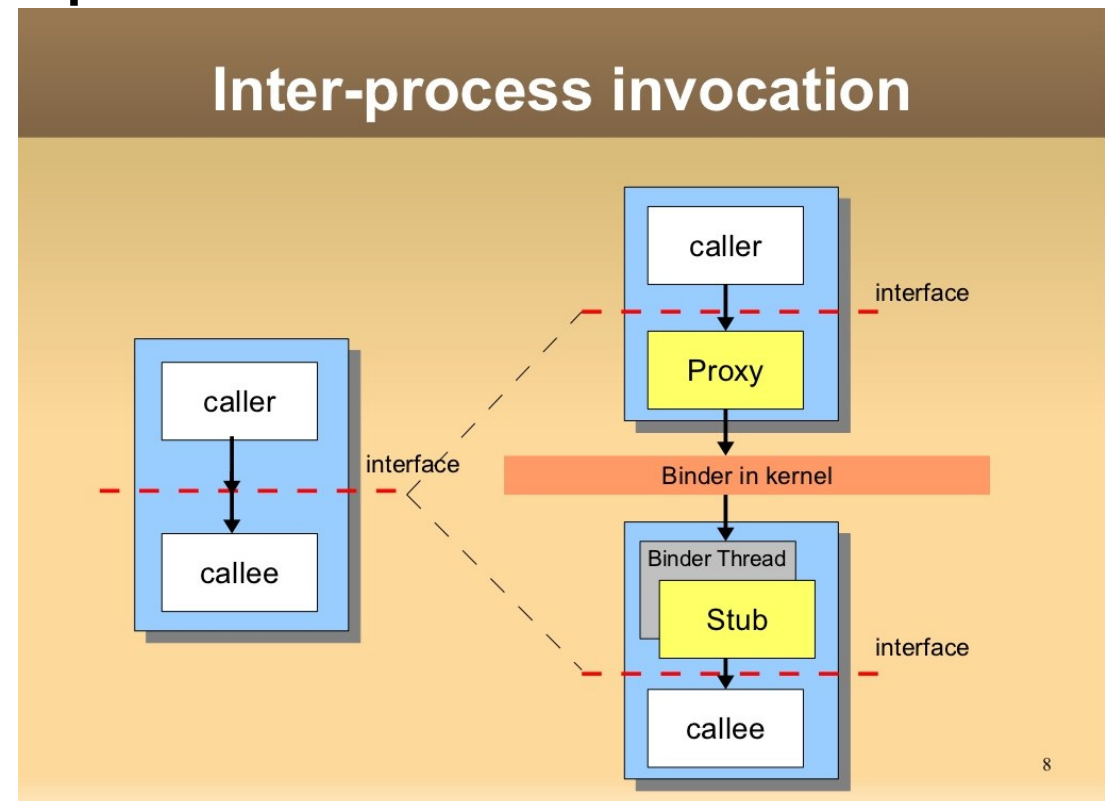
- Els processos s'eliminen si...
 - Són matats per una altra aplicació
 - `public static final void killProcess (int pid)`
 - El sistema es troba amb poca memòria lliure
 - Es selecciona el procés amb més *oom_score* i es mata
 - Fet a través d'un driver de Linux
 - Lowmemkiller

Afegits d'Android a Linux

- Binder Android IPC
- Ashmem Android shared memory
- PMEM Proces memory allocator
- logger Android Logger
- wakelock Android Power Management
- Low Memory Killer Out-of-memory handling
- Alarm Android Alarm
- ... a large number of minor patches

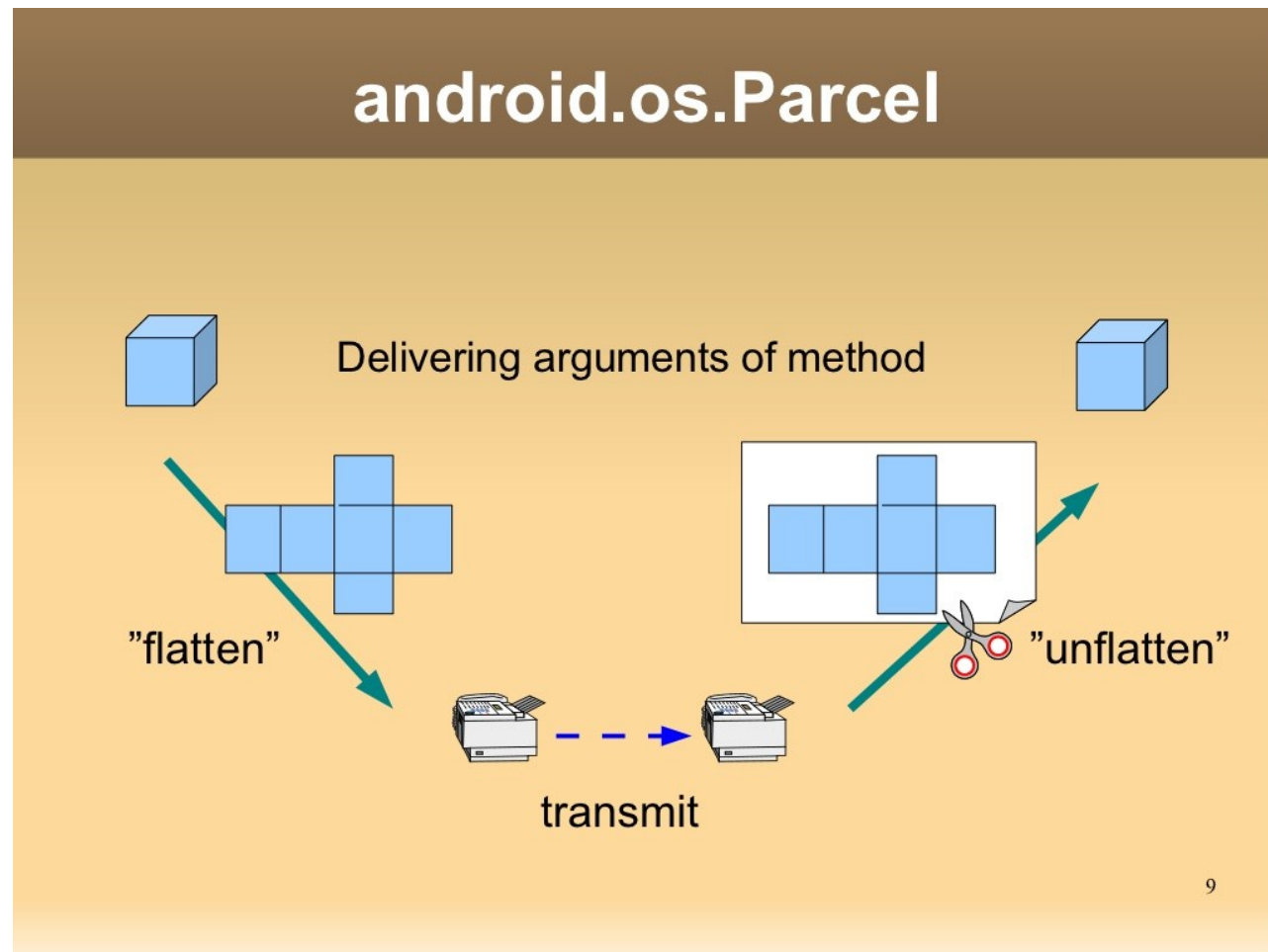
Binder IPC

- Permet que un procés invoqui mètodes en l'espai d'un altre procés
- Binder permet identificar el mètode a invocar i poder passar-li els paràmetres



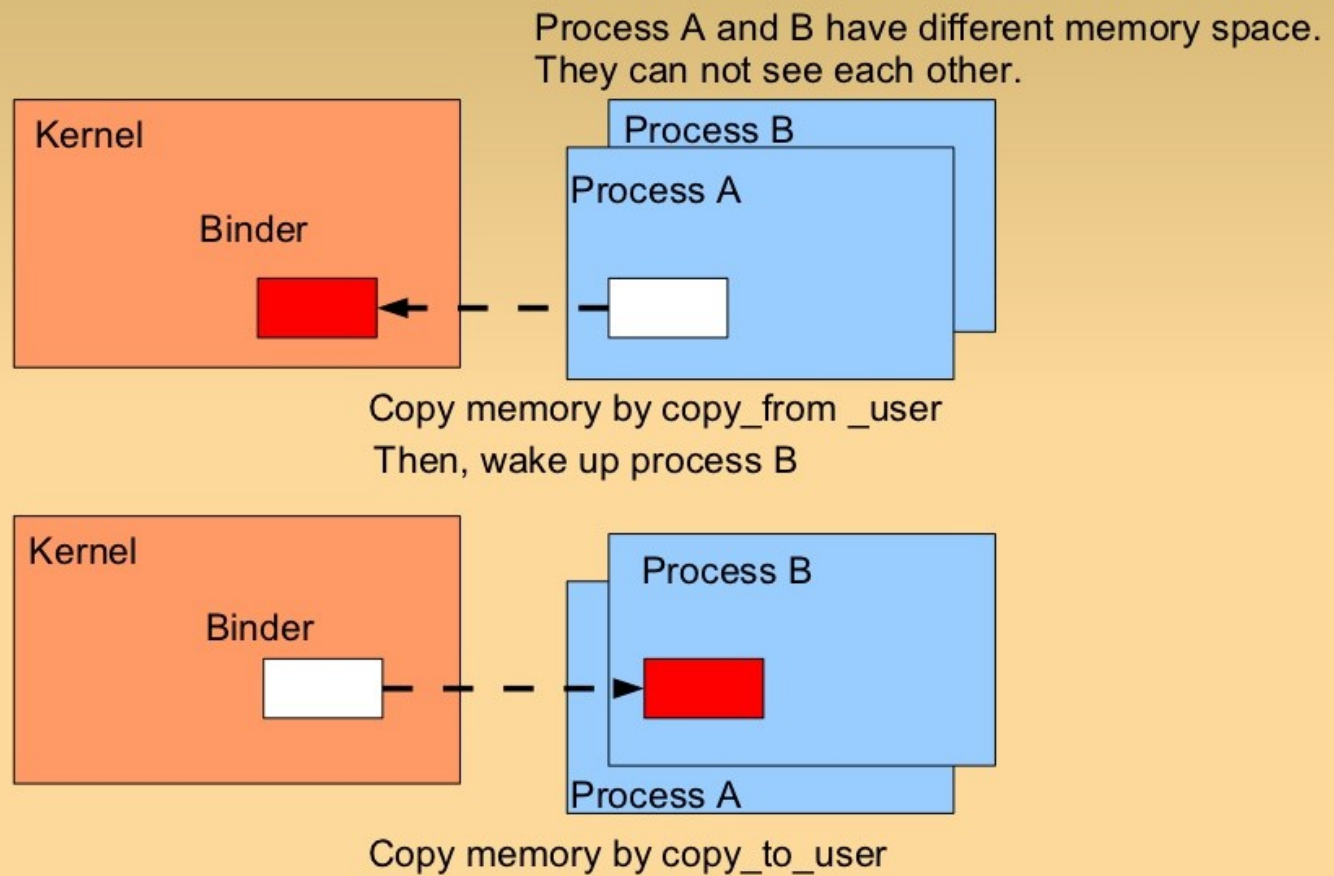
Binder IPC

- Marshaling / unmarshaling
- Serialization / deserialization
- Permet passar objectes
 - Descriptors de fitxer...
- RPCGEN+ (?)



Binder IPC

Transaction of Binder



Ashmem (shared memory)

- "Shared memory allocator, similar to POSIX SHM"
 - Different behaviour
 - Exporting a simpler file-based API
- Característica:
 - Pot descartar regions de memòria compartida en situacions de poca disponibilitat de memòria

```
// Establir la regió de memòria compartida
fd = ashmem_create_region("my_shm_region", size);
if(fd >= 0) {
    data = mmap(NULL, size, PROT_READ | PROT_WRITE, MAP_SHARED, fd, 0);
}
if(fd < 0 || data == MAP_FAILED)
    err (1, "Unable to create shared memory region");
```

Ashmem (shared memory)

- Per compartir memòria el procés creador ha de passar el descriptor de fitxer (fd)
 - A través del binder
- Un cop passat, la regió es mantindrà mentre hi hagi processos que la mantinguin oberta: responsabilitat compartida
- Com? `dup(fd)` entre processos no relacionats
 - `writeFileDescriptor(FileDescriptor)`
 - `readFileDescriptor(FileDescriptor)`
 - El procés destí rep un `dup(fd)` i els dos processos comparteixen el punter de lectura/escriptura

PMEM – Process memory allocator

- Gestiona memòria compartida entre l'usuari i els dispositius (dsp, gpu...)
 - 1 – 16+ MB
 - **físicament contigua**
 - necessari per limitacions en certes màquines
 - no cal en x86
 - el creador ha de mantenir l'àrea mentre hi hagi altres processos accedint-hi: responsabilitat centralitzada

Logger

- Suport en el kernel per la comanda *logcat*
- Equivalent al *dmesg*, però pels processos de sistema
 - Dalvik VM
 - Activity Manager
 - ...

```
W/ActivityManager( 280): Unbind failed: could not find connection for android.os.BinderProxy@4105bd28
I/Choreographer( 648): Skipped 45 frames! The application may be doing too much work on its main thread.
D/dalvikvm( 662): GC_CONCURRENT freed 462K, 19% free 2472K/3052K, paused 67ms+3ms, total 718ms
W/EGL_emulation( 401): eglSurfaceAttrib not implemented
E/ThrottleService( 280): problem during onPollAlarm: java.lang.IllegalStateException: problem parsing stats:
java.io.FileNotFoundException: /proc/net/xt_qtaguid/iface_stat_all: open failed: ENOENT (No such file or
directory)
I/ActivityManager( 280): No longer want android.process.media (pid 492): empty for 3645s
```

Wakelocks

- Gestió del consum
- Mecanisme que impedeix que el sistema entri en un estat de baix consum
- Kernel
 - `wake_lock_init (wakelock *, type, name *)`;
 - `WAKE_LOCK_SUSPEND...` impedeix suspend
 - `WAKE_LOCK_IDLE...` impedeix low-power idle states
 - `wake_lock (wakelock *)`;
 - `wake_lock_timeout (wakelock *, timeout)`;
 - `wake_unlock (wakelock *)`;
- Usuari
 - Escriure un nom a `/sys/power/wake_lock` i a `/sys/power/wake_unlock`

<http://lwn.net/Articles/318611/> "Wakelocks and the embedded problem", Jonathan Corbet

Wakelocks

- Crítiques que han impedit que entrés a *mainline*
 - Duplicat de pm_qos
 - Power management quality of service
 - No s'ha fet cap esforç per integrar la mateixa funcionalitat a pm_qos
 - Sembla que es podria implementar amb un comptador
 - Incompatible amb /sys/power/state
 - No hi ha recuperació possible si un procés acaba mentre té un wakelock !!
 - El comportament per defecte del sistema és *suspend*, encara que hi hagi processos executant-se
 - No és bo per servidors!!!

Wakelocks

- En Linux es mou un suport similar
 - La infraestructura pm_stay_awake / pm_relax s'ha incorporat al kernel
 - Similar als wakelocks
 - No està clar que sigui suficient per Android

drivers/misc/lowmemorykiller.c

security/lowmem.c

Low Memory Killer

- Mata processos quan la quantitat de memòria lliure és baixa
 - Evita observar els efectes del *thrashing*
 - Processos ordenats per oom_adj
 - -17 a +15 (?)
 - Alt score significa més opcions de ser matat en cas de manca de memòria
 - OOM-killer (Linux) (?)

```
/proc/3662/oom_score
```

```
39
```

```
$ ps -ef | grep 3662
```

```
xavim      3662  2177  1 Nov12 ?
```

```
00:55:05 /usr/bin/firefox
```

```
$ date
```

```
Thu Nov 14 17:37:42 CET 2013
```


Alarm timers

- Suport per l'AlarmManager
 - Permet que una aplicació s'invoqui a una hora determinada
 - Posteriorment es torna a invocar amb un Intent Obj.
- Permeten que el dispositiu sencerc entri en mode *suspend* i l'alarma sigui capaç de despertar-lo

<http://developer.android.com/about/versions/android-4.4.html>

"To improve power efficiency, Android now **batches together alarms** from all apps that occur at reasonably similar times so the system wakes the device once instead of several times to handle each alarm."

Android *suspend/resume*

- Suspend / hibernate redueixen el consum
 - Juntament amb reduir consum en idle
 - Halt...
- Temps de *resume* decreixent
- Dificultats
 - ... si s'espera alguna funcionalitat activa en el sistema
 - ... cal millorar el suport del RTC i la seva interfície

Android *suspend/resume*

- Disseny
 - El sistema intenta fer el *suspend* sempre
 - Les aplicacions i el kernel l'impedeixen amb els wakelocks
 - Si ningú té cap wakelock, el sistema entra en *suspend*
 - Abans de suspendre's, es programa l'RTC per provocar una interrupció en el moment en que hi hagi programada la primera alarma
 - Despertador
 - Tasca programada
 - Avís de l'agenda
 - Trucada entrant (?)

Linux alarms

- Funcionalitat disponible
 - Alarma que treu el PC/laptop de *suspend*
 - Comanda `rtcwake`

RTCWAKE(8)

System Administration

RTCWAKE(8)

NAME

`rtcwake` - enter a system sleep state until specified wakeup time

SYNOPSIS

```
rtcwake [-hvVluan] [-d device] [-m standby_mode] {-t time_t|-s seconds}
```

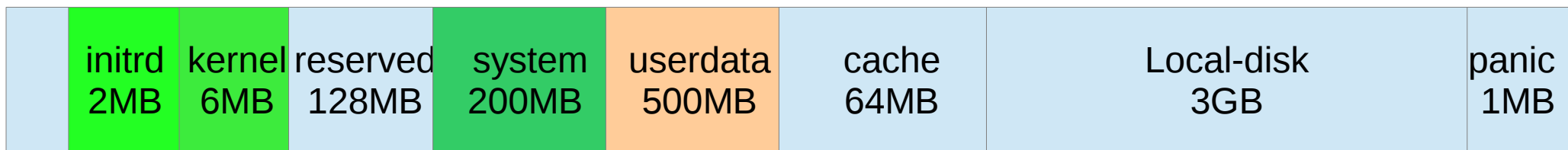
DESCRIPTION

This program is used to enter a system sleep state until specified wakeup time.

This uses cross-platform Linux interfaces to enter a system sleep state, and leave it no later than a specified time. It uses any RTC framework driver that supports standard driver model wakeup flags.

Procés de boot

- Molts sistemes Android usen U-Boot com a boot loader
 - Fa la funció de la BIOS dels PCs
 - U-Boot sap carregar el kernel i l'initrd de la SDcard (MTD flash) interna del dispositiu



- Si el dispositiu permet actualitzar el software a través d'un paquet d'upgrade conegut, obre la porta a atacs

<http://www.denx.de/wiki/U-Boot/WebHome>

Sistemes de fitxers

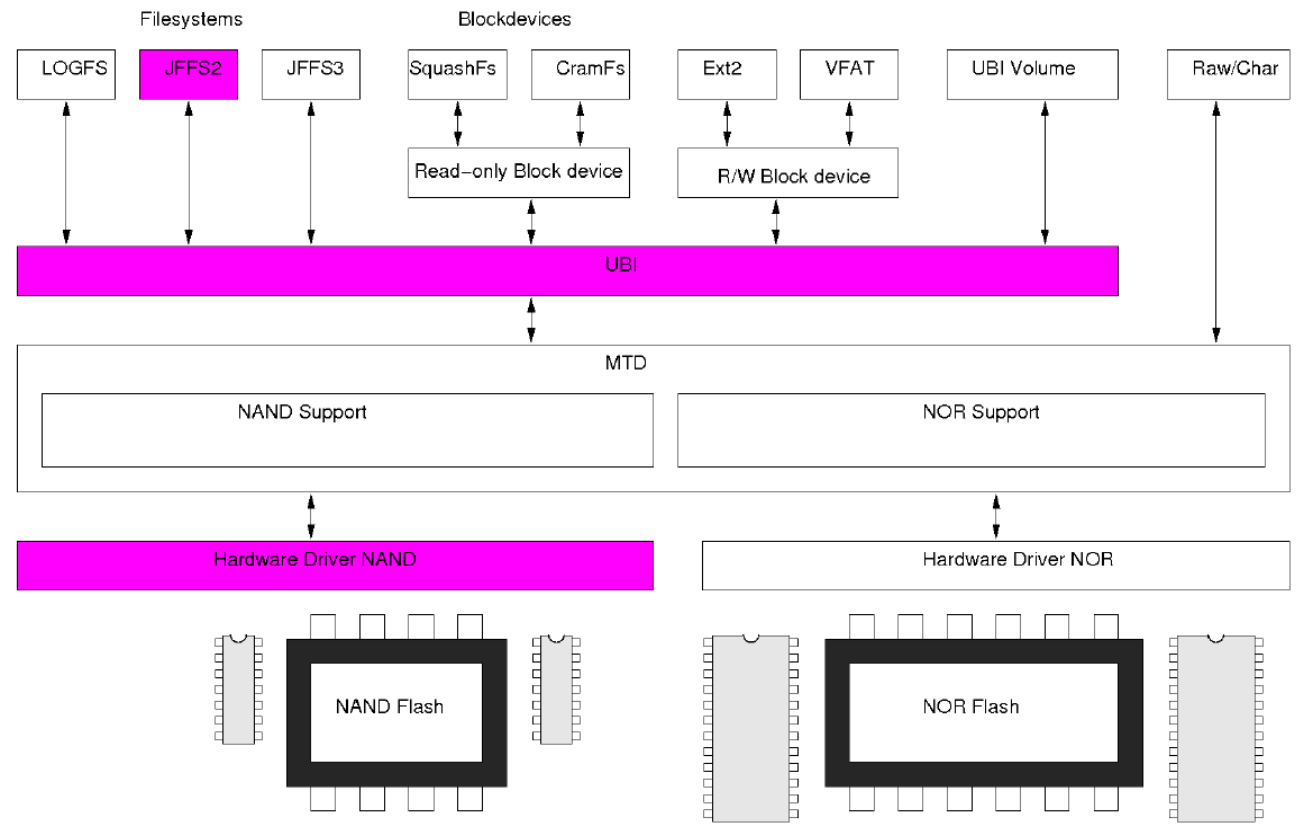
- Unsorted Block Images – UBI

- Proporcionen suport a dispositius MTD interns

- Memory Technology Device

- NAND, NOR flash, amb 3 operacions: llegir, escriure i esborrar

- No per tarjetes de memòria



Sistemes de fitxers

- MTD / UBI
 - Les memòries flash poden tenir errors
 - Bit flips
 - Habitualment corregits per ECC checksums
 - Però al cap del temps cal reemplaçar el bloc per un de nou -> UBI (scrubbing, de forma transparent)
 - Proporciona volums que poden ser dinàmicament
 - creats, esborrats o canviats de mida
 - Distribueix les escriptures per equilibrar la degradació dels blocs

Sistemes de fitxers

<http://teamw.in/exFAT>

- exFAT – extended file allocation table
 - Disenyat per Microsoft, a partir del FAT32
 - Flexible Intellectual Property (IP) licensing program
 - We'll probably never see exFAT support in the Linux kernel
 - Proprietary and patent-pending
 - Adoptat per
 - SD Card Association, SDXC cards
 - Memory Stick XC™ Series
 - Proveidors d'Android tenen acords amb Microsoft que els permeten proporcionar el suport
 - Alternatives: ext4, NTFS

Sistemes de fitxers

- exFAT – extended file allocation table
 - Afegeix suport per dispositius de 32 Gb. a 256 TB.
 - Capacitats
 - Fitxers més grans de 4 Gb.
 - 4000 imatges RAW, 100 pel·licules HD, 60 hores de gravació HD
 - Interoperabilitat amb altres OSs (OS X)
 - Extensible
 - OEMs poden afegir les seves pròpies característiques al sistema de fitxers
 - Possibilitat de journaling (transactions): TexFAT

Sistemes de fitxers

- exFAT per Linux implementat amb FUSE
- Samsung posa la seva versió en GPL
 - a GitHub (<http://github.com>)
 - Portat d'Android

Programació d'aplicacions

- Creació d'un nou projecte

- android list targets id: 1 or "android-17"
- android create avd -n <name> -t <targetID> [-<option> <value>]
id: 2 or "Google Inc.:Google APIs:17"
- emulator -avd <avd_name> [<options>]
- emulator -avd <avd_name> -gpu on # experimental
- android create project --target android-17 --name MyFirstApp --path workspace/MyFirstApp --activity MainActivity --package com.example.myfirstapp

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

Programació d'aplicacions

- Project contents
 - build.xml: regles de construcció de l'app
 - local.properties: path to the Android SDK
 - project.properties: target interface (android-17)
 - ant.properties: referència a la clau per signar l'app
 - AndroidManifest.xml: requeriments
 - Exemple: `<uses-feature android:glEsVersion="0x00020000" android:required="true" />`
 - src: codi font, Java
 - jni: codi font, C/C++
 - res: fitxers usats per l'app (gràfics, text, la seva composició...)
 - libs: llibreries privades d'aquesta app
 - gen: fitxers generats per la compilació, interfícies
 - bin: resultat de la compilació (*.apk...)

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

Properties files

- `local.properties` `sdk.dir=/mnt/opt/xavim/Android4/android-sdk-linux`
- `project.properties` `target=android-17`
- `res`
 - `drawable (icona...)`
 - `layout`
 - `values/strings.xml` `<string name="hello">Hello, OpenGL ES 2.0!</string>`

Programació d'aplicacions

- Compilació del Java sobre Dalvik
 - `ant {release | debug }` # Apache Ant (make)
- Instal·lació
 - `[emu] adb connect localhost`
 - `adb install bin/Benchmark-debug.apk`
 - en mode debug, no cal signar les apps
 - l'aplicació es copia al dispositiu
 - accessible per la seva icona

Programació d'aplicacions

- Accés als serveis nadius (jni)
 - NDK
 - cd <any-sample or project>
 - \$NDK/ndk-build
 - android update project -p . --target android-17
 - ant debug

Interfície de threads

- Agafar i posar prioritat
 - get/set ThreadPriority
- Enviar signals, acabar (kill)
 - sendSignal(pid, signal)
 - SIGNAL_KILL, QUIT, USR1
- Agafar pid, threadid, els temps de cpu, l'uid...
 - myPid(), getElapsedCpuTime()

Tasques asíncrones

- Executades en un sol thread per evitar errors de programació paral·lela
 - Per executar en paral·lel, cal indicar-ho
- S'invoquen un col·loc i segueixen 4 passos
 - Execució prèvia: `onPreExecute()`
 - Inici de la tasca: `doInBackground(arguments...)`
 - Indicació de progrés: `onProgressUpdate (...)`
 - Execució posterior: `onPostExecute(resultat)`

Control del temps

- Comptador d'alarma
 - CountdownTimer
 - start, cancel, onTick
 - SystemClock
 - elapsedRealttime, milisecons des del boot
 - sleep (ms)
 - uptimeMillis → uptime

File system statistics

- StatFs (df)
 - Permet accedir a les estadístiques del sistema de fitxers
 - Número total de blocs
 - Blocs disponibles
 - Mida del bloc

Entrada/sortida a fitxers

- Segueix les classes de Java
 - BufferedReader/Writer
 - Per strings
 - InputStream / OutputStream
 - Per bytes, sense tipus
 - DataInputStream / DataOutputStream
 - Per dades amb tipus (big endian)
 - File
 - Classe bàsica per gestionar path i filenames

Entrada/sortida a fitxers

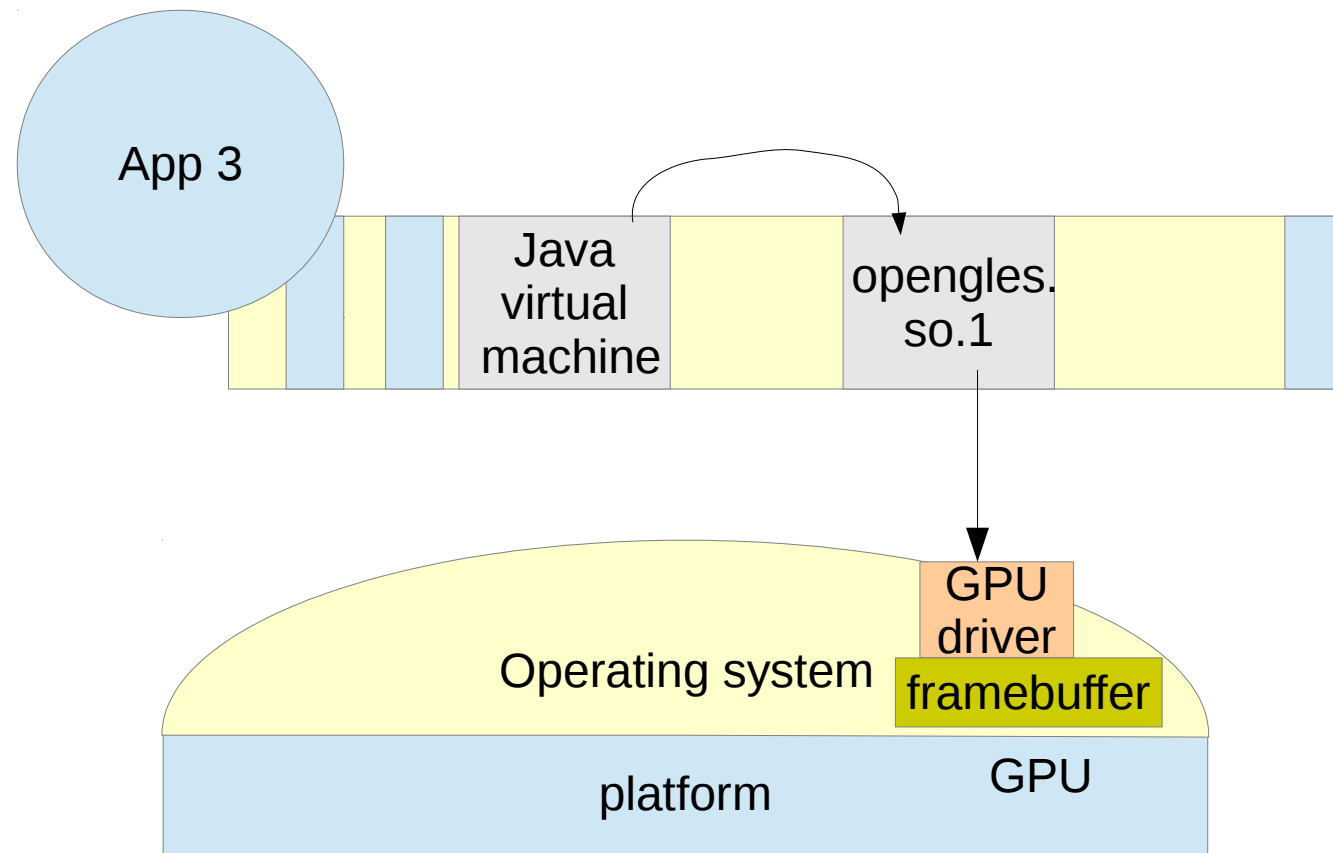
- `DataOutputStream`
 - `write...` `String`, `char`, `float`, `double`, `int...`
 - `size()`: quants bytes s'han escrit al fitxer
 - `flush()`, `close()`

Comunicacions

- Sobre sockets de Java
 - `java.net.DatagramSocketImpl`
 - Datagrama, UDP, sense control d'errors
 - `java.net.SocketImpl`
 - Stream, TCP, amb control d'errors
 - `bind`, `listen`, `accept`, `connect`

Gràfics

- Sobre OpenGL ES (Embedded GL)
 - Versió reduïda d'OpenGL



Sincronització

<http://developer.android.com/reference/android/os/ConditionVariable.html>

- Variables de condició
 - close(), tanca la regió
 - block(), bloqueja el flux, si la regió està tancada
 - open(), obre la regió, despertant tots els fluxos
- Mutex, només un flux entra a la regió crítica
 - acquire
 - release

Monitorització

- Android device monitor
- Interfície: Android Debug Bridge – adb
- Permet seguir l'execució en el dispositiu
 - Fluxos
 - Heap
 - Objectes
- Permet també obtenir una shell del dispositiu
 - Toolbox / busybox + eines diverses de Linux

Treball personal

- Fes un informe explicant el
 - SSH Filesystem
 - Quantes versions hi ha hagut?
 - Com funcionen?
 - Quina és la que es distribueix actualment?