

MAS: Betriebssysteme

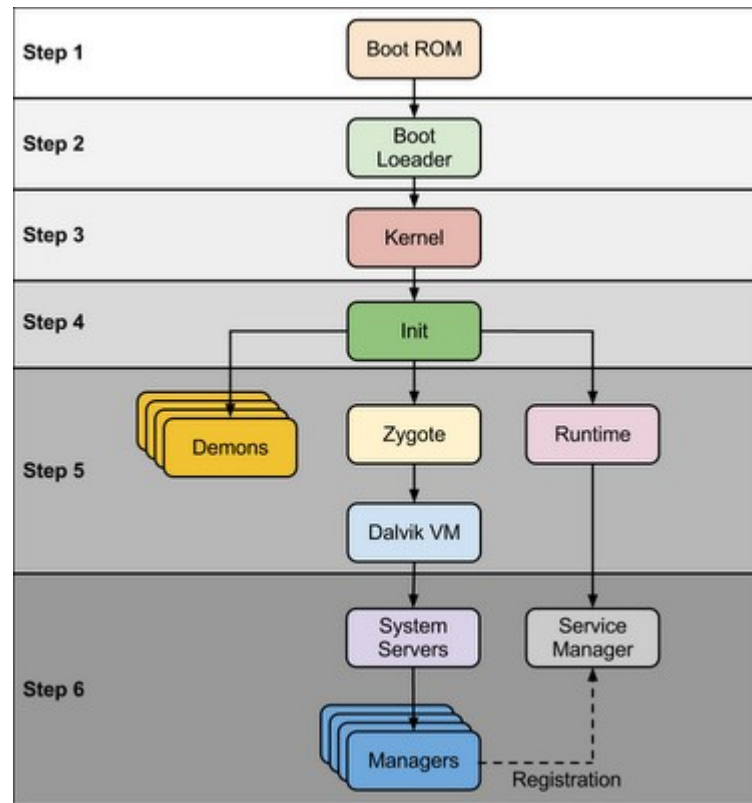
Android Architektur

T. Pospíšek

Hackable? Open Source?

- Q: Kann ich alles was hier folgt nachvollziehen, Android unter die Haube schauen, Zeugs ändern?
- A: Vielleicht:
 - grundsätzlich wäre Android Open Source...
 - wenn man Oberes tun will, muss man beim Kauf eines Handys das Kriterium «*Freiheit*» berücksichtigen - oder Glück haben
 - einige Hersteller erlauben es, durch ein mehr oder weniger mühseliges Prozedere «*root Rechte*» auf dem Gerät zu erlangen
 - dabei muss man u.U. die Geräte-Garantie aufgeben
 - andere Geräte können «*gehackt*» werden
 - Legal? Garantie? Achtung!

Boot



Boot

- 1) proprietärer Bootloader vom Hersteller
 - initialisiert Minimum an HW
 - lädt ...
- 2) Linux Kernel evtl. mit Ramdisk/Image
 - initialisiert HW
 - startet ...
- 3) init (Android-eigenen)
 - führt /init.rc aus
 - konfiguriert:
 - Userspace (Rechte)
 - Hardware
 - bindet Dateisysteme ein
 - startet:
 - Daemons/low level Services
 - Zygote

Zygote

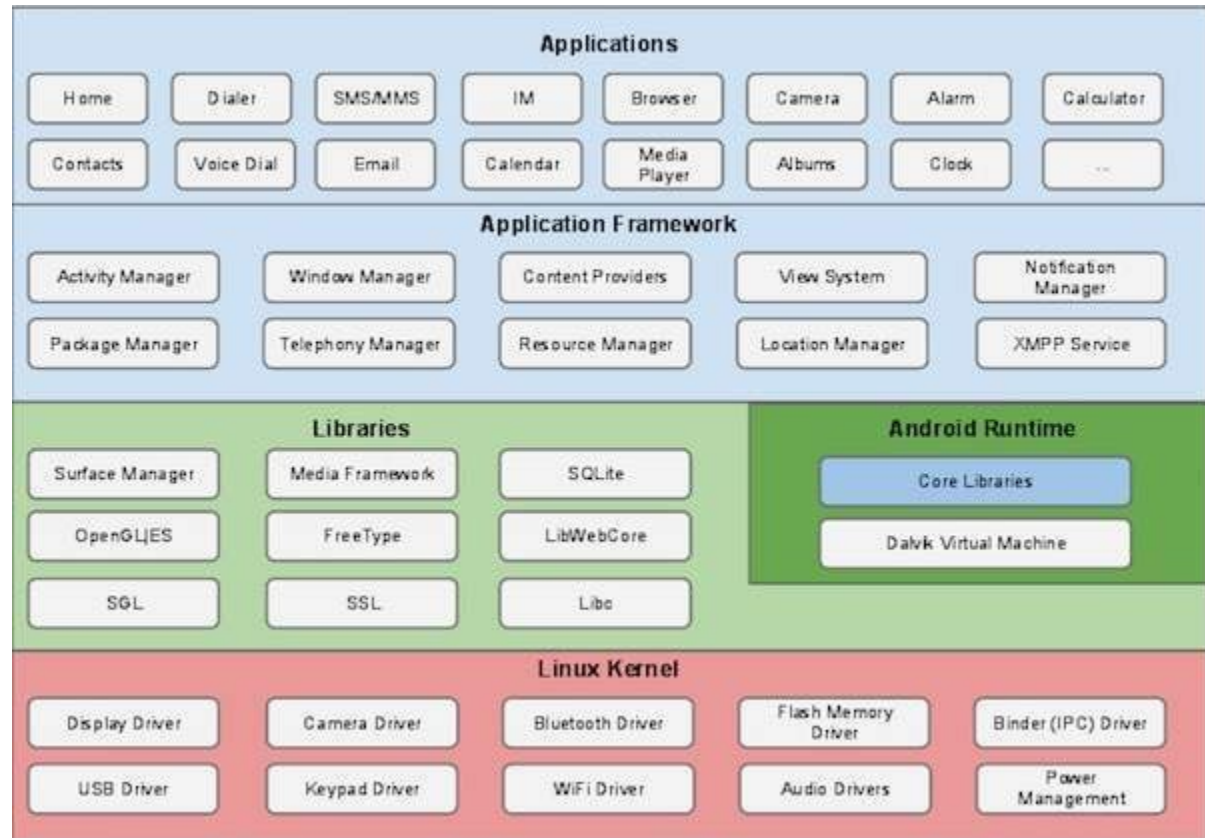
- Zygote ist der «Init Prozess der Android Schicht»
 - startet eine Dalvik JVM, welche die Klassen des Android Runtime Frameworks geladen hat
 - Zygote forkt eine VM weg, welche wiederum die Android Services startet und Android Applikationen starten kann (also alles was man als «Android» kennt)
 - neue Applikationen oder Services werden erstellt indem Zygote angefragt wird und einen neuen Prozess (mit vorinitialisiertem Dalvik) wegforkt

Dalvik

- die Dalvik VM ist sparsam bzgl. Speicherverbrauch, indem Klassen und die VM selbst read-only sind und mit copy-on-write zwischen Prozessen geteilt werden

Schichten Model

- neuere Androids haben eine zusätzliche HAL Schicht



Quelle: <https://www.tutorialspoint.com/android/images/architecture.jpg>

Android Runtime

Applikations-Komponenten

- Activity → Activity Manager
 - z.B. :
 - Email anzeigen
 - Email schreiben
 - Emails auflisten
 - eine Activity ist die MainActivity und somit der Einstiegspunkt der Applikation
- Service
 - läuft im Hintergrund ab
 - kann auch laufen, wenn die App, welche ihn gestartet hat nicht mehr läuft

Android Runtime

Applikations-Komponenten

- Receiver / BroadcastReceiver
 - hört auf Nachrichten von Applikationen oder von System
 - definiert Reaktionen auf Nachrichten («Intents»)

- Provider / ContentProvider
 - Daten einer anderen Applikation zur Verfügung stellen auf Anfrage

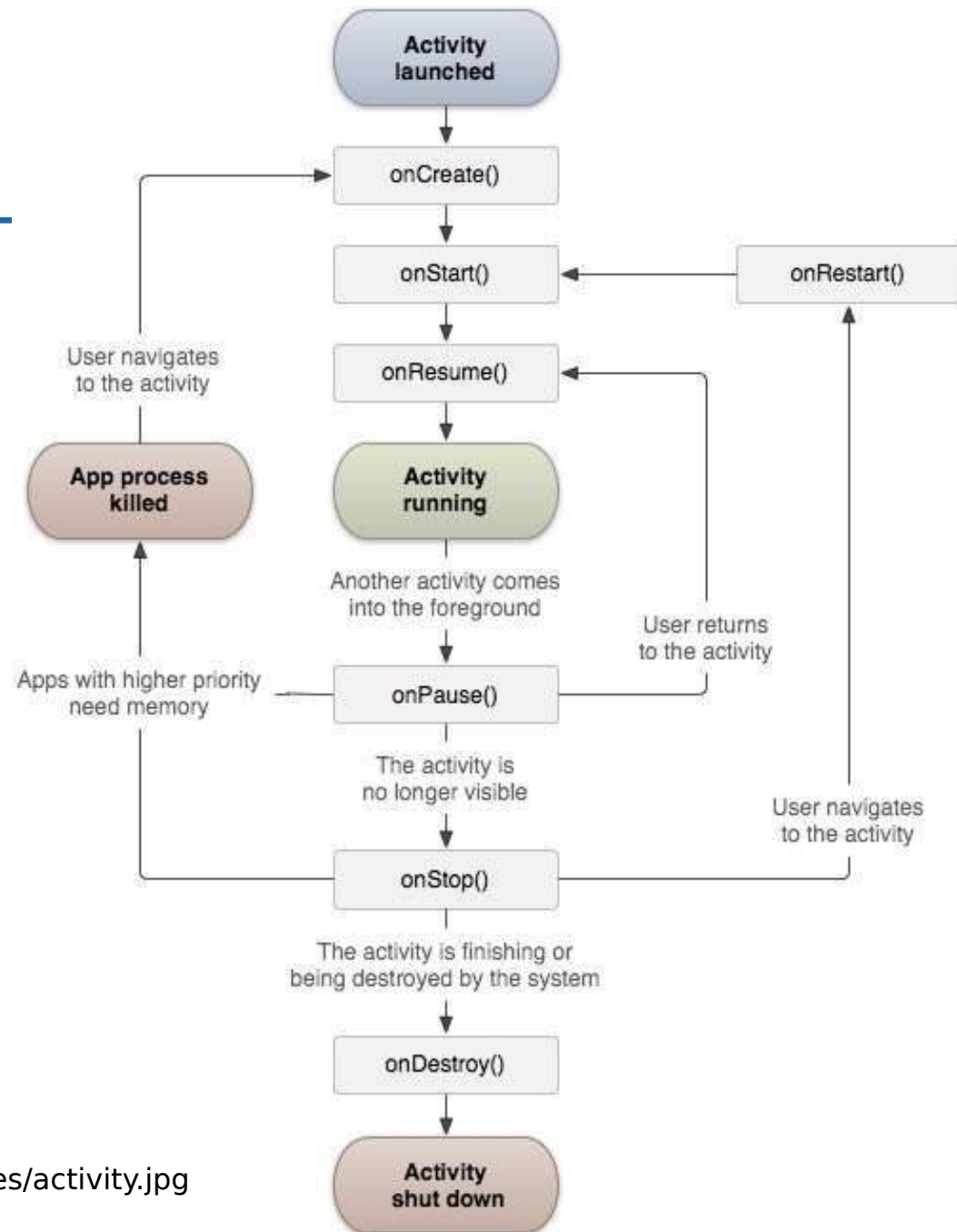
Android Runtime

Applikations-Komponenten

- manifest.xml
 - deklariert die verfügbaren Komponenten der Applikation für den Applikations-Launcher (Zygote/Dalvik)
 - dadurch weiss der Launcher beim Start der Applikation, was er mit der Applikation machen muss, welche Ressourcen er zur Verfügung stellen muss

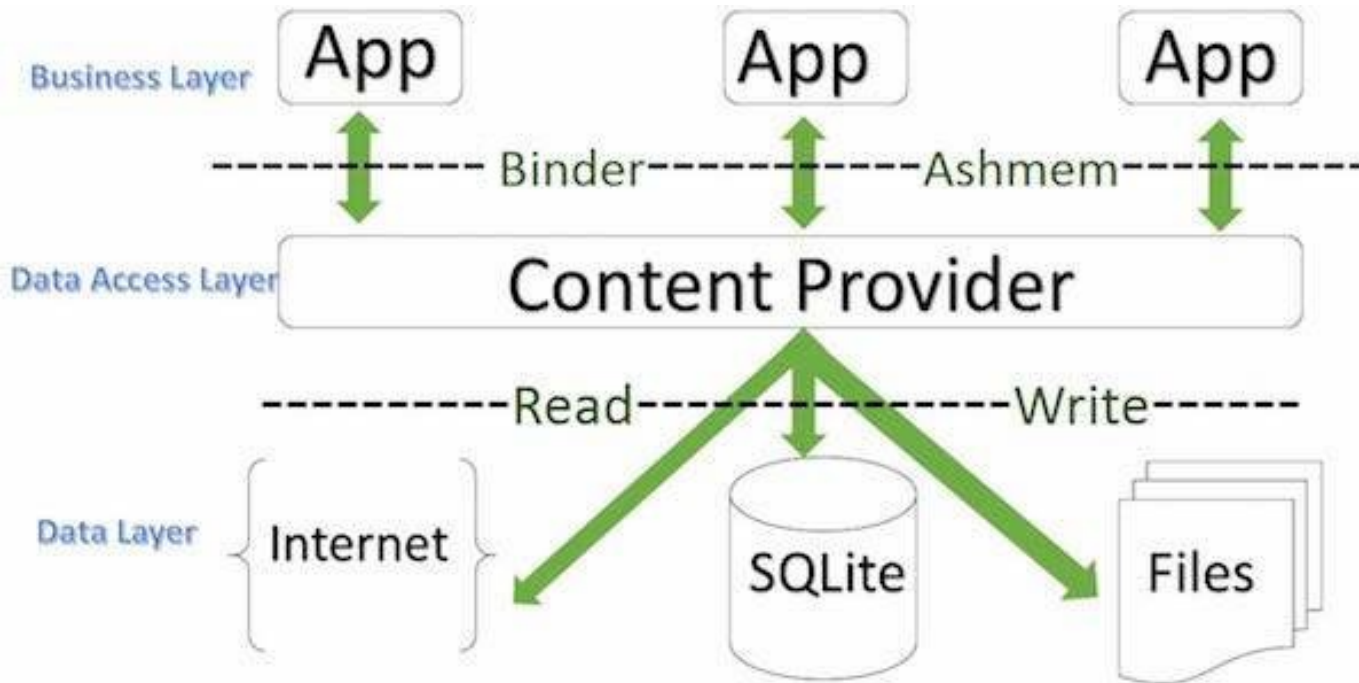
Android Runtime Applikations-Lifecycle

- *onIrgendwas()* sind Callbacks, welche die App deklarieren kann, um auf Zustandsänderungen reagieren zu können



Quelle:
<https://www.tutorialspoint.com/android/images/activity.jpg>

Android Content Provider



Quelle: <https://www.tutorialspoint.com/android/images/content.jpg>

- «REST-ful» Data
- `content://provider/daten_typ/id`

Android Intents

- Intents
 - sind der IPC / RPC Mechanismus von Android
 - deklarieren Aktionen, die ausgeführt werden sollen:
 - Aktion + Daten
 - werden z.B. übergeben bei:
 - Context.startActivity()
 - Context.startService()
 - Context.sendBroadcast()
- Beispiel eines Intents (Aktion, Daten):
 - (ACTION_DIAL, content://contacts/people/1, Optionen)

Referenzen

- http://rts.lab.asu.edu/web_438/project_final/CSE_598_Android_Architecture_Binder.pdf
- https://www.tutorialspoint.com/android/android_architecture.htm
- https://elinux.org/Android_Portal
- https://elinux.org/Android_Booting
- <https://source.android.com/devices/bootloader>
- <http://www.onsandroid.com/2014/10/in-depth-android-boot-sequence-process.html>
- https://android.googlesource.com/platform/system/core/+/_android-4.1.2_r1/init/readme.txt
- <https://stackoverflow.com/questions/9153166/understanding-android-zygote-and-dalvikvm>