Pázmány Péter Catholic University
Faculty of Information Technology and Bionics

Android Development

Android System

Android FS

File system

- Android is linux based
  - Therefore the file system is similar to common linux systems
  - Typical mount points

File system

```
/
/dev
/dev/pts
/proc
/sys
/sys/fs/selinux
/mnt
/metadata
/vendor
/product
/sys/kernel/debug
/mnt/vendor/persist
/config
/vendor/firmware_mnt
/storage
/data
/mnt/runtime/default/emulated
/storage/emulated
```

In case of Nexus 6P

- The output of the df (diskfree) command

```
df
Filesystem     1K-blocks     Used Available Use% Mounted on
tmpfs            1425056      444   1424612   1% /dev
tmpfs            1425056        0   1425056   0% /mnt
/dev/block/dm-0  2999516  2660328    322804  90% /system
/dev/block/dm-1   194280   190400         0 100% /vendor
/cache             96688     5320     89320   6% /cache
/modem             81872    48688     33184  60% /firmware
/dev/block/dm-2 26225216 14459708  11749124  56% /data
/dev/fuse       26225216 14459708  11749124  56% /storage/emulated
```

In case of Pixel 4 XL

- The output of the df (diskfree) command

```
coral:/ $ df -aH
Filesystem       Size  Used Avail Use% Mounted on
/dev/block/dm-6  793M  790M     0 100% /
tmpfs            2.8G  795k  2.8G   1% /dev
devpts              0     0     0   0% /dev/pts
proc                0     0     0   0% /proc
sysfs               0     0     0   0% /sys
selinuxfs           0     0     0   0% /sys/fs/selinux
tmpfs            2.8G     0  2.8G   0% /mnt
tmpfs            2.8G     0  2.8G   0% /apex
/dev/block/dm-7  731M  729M     0 100% /vendor
/dev/block/dm-8  2.2G  2.2G     0 100% /product
debugfs             0     0     0   0% /sys/kernel/debug
none                0     0     0   0% /config
bpf                 0     0     0   0% /sys/fs/bpf
tracefs             0     0     0   0% /sys/kernel/debug/tracing
tmpfs            2.8G     0  2.8G   0% /storage
/dev/block/dm-9   54G   35G   18G  67% /data
/data/media       54G   35G   18G  67% /storage/emulated
pstore              0     0     0   0% /sys/fs/pstore
```

File system – Pixel 4 XL

```
dr-xr-xr-x 250 root   root       0 1970-06-11 17:37 acct
drwxr-xr-x  17 root   root     340 2020-04-23 20:59 apex
lrw-r--r--   1 root   root      11 2009-01-01 01:00 bin -> /system/bin
lrw-r--r--   1 root   root      50 2009-01-01 01:00 bugreports ->
/data/user_de/0/com.android.shell/files/bugreports
lrw-r--r--   1 root   root      19 2009-01-01 01:00 charger -> /system/bin/charger
drwxr-xr-x   4 root   root       0 1970-01-01 01:00 config
lrw-r--r--   1 root   root      17 2009-01-01 01:00 d -> /sys/kernel/debug
drwxrwx--x  46 system system  4096 2020-04-23 20:59 data
drwxr-xr-x   2 root   root    4096 2009-01-01 01:00 debug_ramdisk
lrw-------   1 root   root      23 2009-01-01 01:00 default.prop -> system/etc/prop.default
drwxr-xr-x  19 root   root    4980 2020-04-30 18:11 dev
lrw-r--r--   1 root   root      15 2009-01-01 01:00 dsp -> /vendor/lib/dsp
lrw-r--r--   1 root   root      11 2009-01-01 01:00 etc -> /system/etc
drwx------   2 root   root   16384 2009-01-01 01:00 lost+found
drwxr-xr-x  12 root   system   260 1970-06-11 17:37 mnt
drwxr-xr-x   2 root   root    4096 2009-01-01 01:00 odm
drwxr-xr-x   2 root   root    4096 2009-01-01 01:00 oem
dr-xr-xr-x 717 root   root       0 1970-01-01 01:00 proc
drwxr-xr-x  14 root   root    4096 2009-01-01 01:00 product
lrw-r--r--   1 root   root      24 2009-01-01 01:00 product_services ->
/system/product_services
drwxr-xr-x   3 root   root    4096 2009-01-01 01:00 res
drwxr-x---   2 root   shell   4096 2009-01-01 01:00 sbin
lrw-r--r--   1 root   root      21 2009-01-01 01:00 sdcard -> /storage/self/primary
drwxr-xr-x   4 root   root      80 2020-04-23 20:59 storage
dr-xr-xr-x  12 root   root       0 1970-06-11 17:37 sys
drwxr-xr-x  12 root   root    4096 2009-01-01 01:00 system
drwxr-xr-x  17 root   shell   4096 2009-01-01 01:00 vendor
```

Properties

- Partitions and mount points (and their sizes) are proprietary to the device manufacturer
  - These are pre-defined for each device
  - It is not impossible to modify
    - It is rare that it is useful
  - The /system mount point contains the factory firmware, the base system
    - The size of this partition may introduce restriction to further upgrades as well
- Some of them is mounted read-only
  - For example /system
    - Thus update of the application of the base system do not replace the file of the firmware as it cannot be overwritten
  - The device has to be rooted to obtain permissions to change
    - Thus in case of the device is rooted, it is possible to replace system components
    - And loose the warranty immediately

Partitions

- `/system`
  - Operating system
    - Linux, ART, …
    - Except for kernel and ramdisk
  - Preinstalled applications
    - Cannot be removed or modified
- `/apex`
  - Application binary libraries that can be updated independently from the system firmware
- `/data`
  - User applications and data
    - Applications downloaded later and updates
    - Application data are stored here
      - Or on SD card
- `/cache`
  - Automatically managed
  - May not exists

Partitions

- `/misc`
  - Special system level settings
  - CID values
  - USB settings
  - Hardware settings
- `/metadata`
  - partition is used when device is encrypted
- `/vendor`
  - binary that is not distributable to the Android Open Source Project (AOSP)
  - if there is no proprietary information, this partition may be omitted.
- `/radio`
  - contains the radio image.

Partitions

- Previously not listed, but very important
- `/boot`
  - Ramdisk and kernel are stored here
  - Without this partition even the system cannot be started
- `/recovery`
  - Alternative boot partition
  - Starting in recovery the firmware upgrade or install can be executed

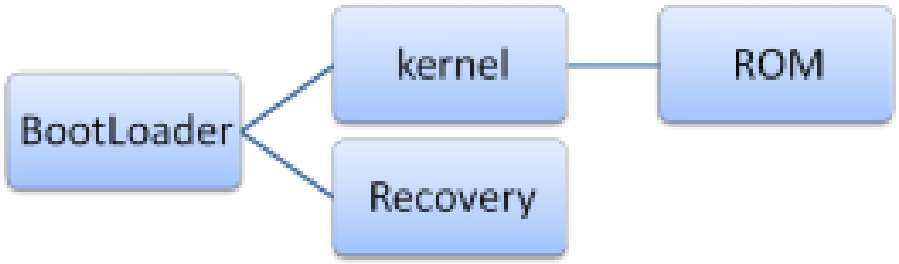
Demonstration – Hands on!

- Discover the partitions of the phone
  - `adb shell`

Boot modes

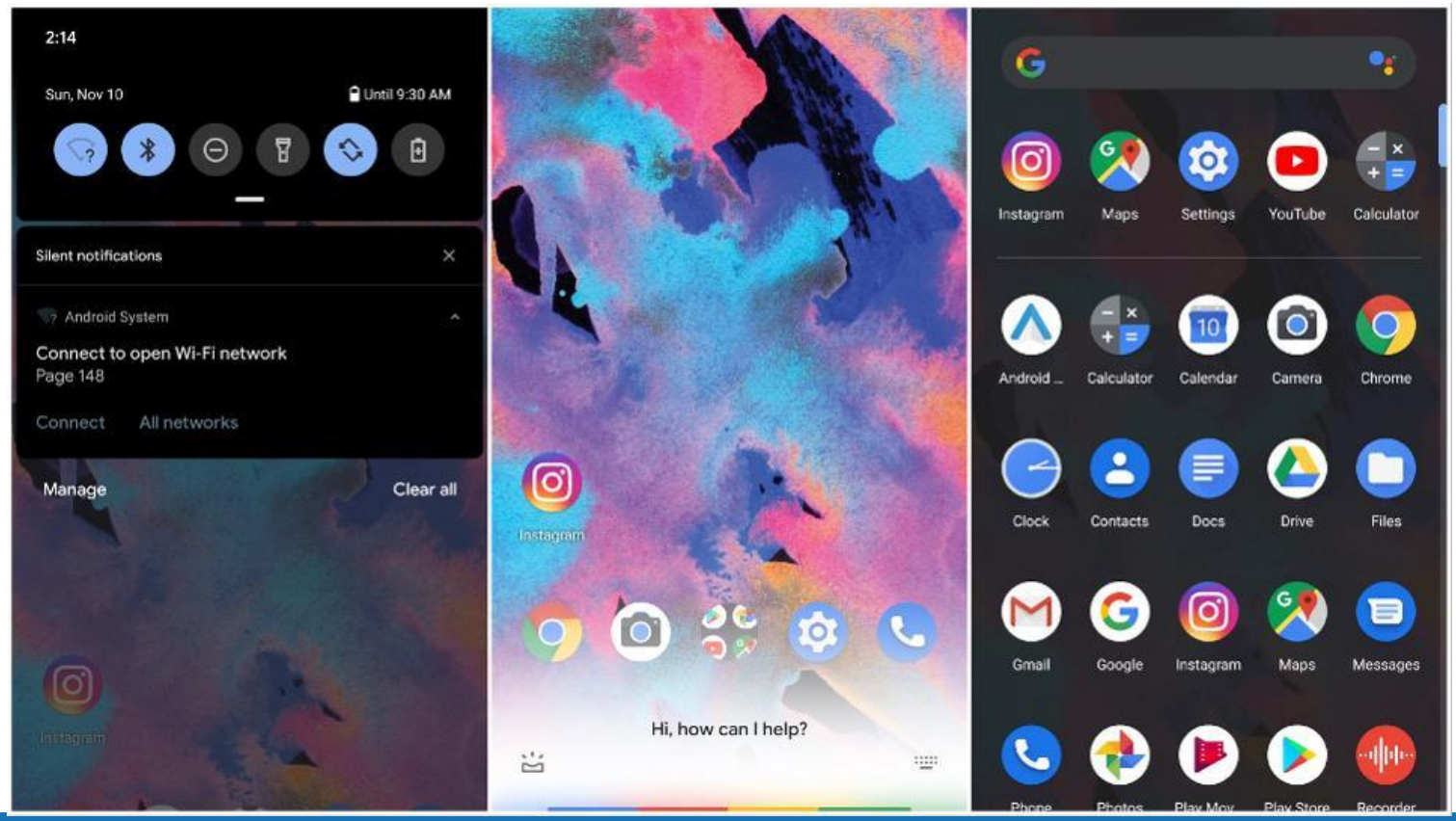
Android modes

- Normal mode – Android kernel
- Recovery mode – Recovery, firmware install
- Each of them is loaded by the Bootloader



Normal boot

- The usual Android kernel is loaded



Bootloader

- The device can be started in bootloader mode by pressing several physical buttons at the same time
  - Or `adb reboot bootloader`
- The possibilities are dependent on the actual bootloader program
  - It is specific to the actual manufacturer
  - Out-of-box it is locked (thus cannot be replaced)
  - However it can be unlocked
    - Using code
    - Hardware tricks
    - Following statement: `fastboot flashing unlock`
- In bootloader mode the adb does not work
  - Instead of the fastboot program can be used
  - In case if unlocked bootloader you can
    - Flash
    - Erase
    - Restart

Bootloader


Start
FASTBOOT MODE
PRODUCT_NAME - hammerhead
VARIANT - HAMMERHEAD d820 32GB
HW VERSION - rev_11
BOOTLOADER VERSION - HHZ11d
BASEBAND VERSION - M894A-1.0.25.0.17
CARRIER INFO - None
SERIAL NUMBER - 030f7083d0210c7d
SIGNING - www.androidxda.com
SECURE BOOT - enable
LOCK STATE - locked

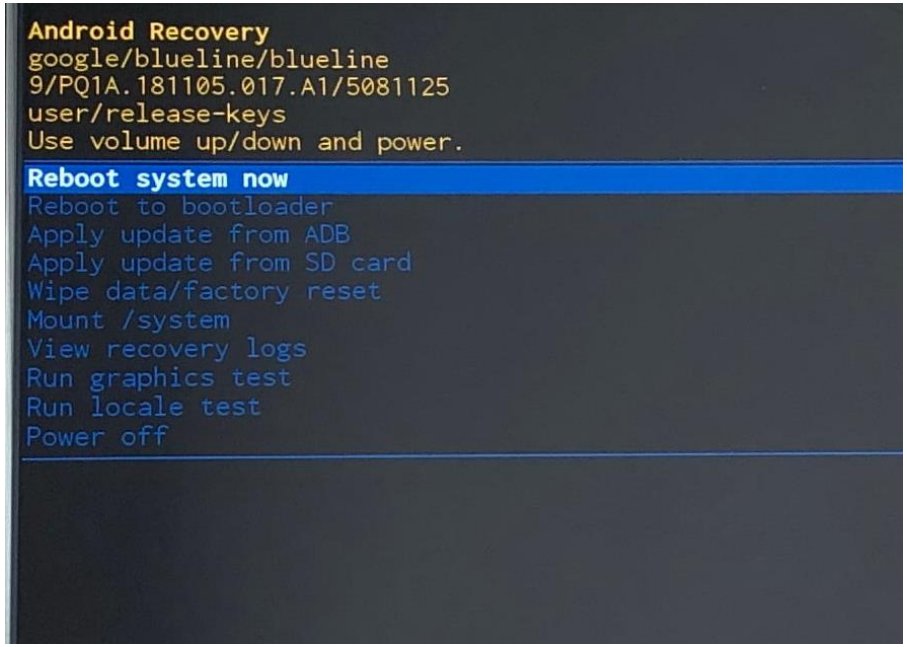

Power off
Press Volume keys to select different menu
Fastboot Mode
Product revision: coral DVT1.0(NA)
Bootloader version: c2f2-0.2-5799621
Baseband version: g8150-00014-191001-B-5910554
Serial number: 96031FFBA001JM
Secure boot: PRODUCTION
NOS production: yes
DRAM: 6GB Micron LPDDR4X
UFS: 128GB Toshiba
Device state: locked
Boot slot: a
Enter reason: combo key
Debug Policy(a): enabled(in-list)
Debug Policy(b): enabled(in-list)

Recovery

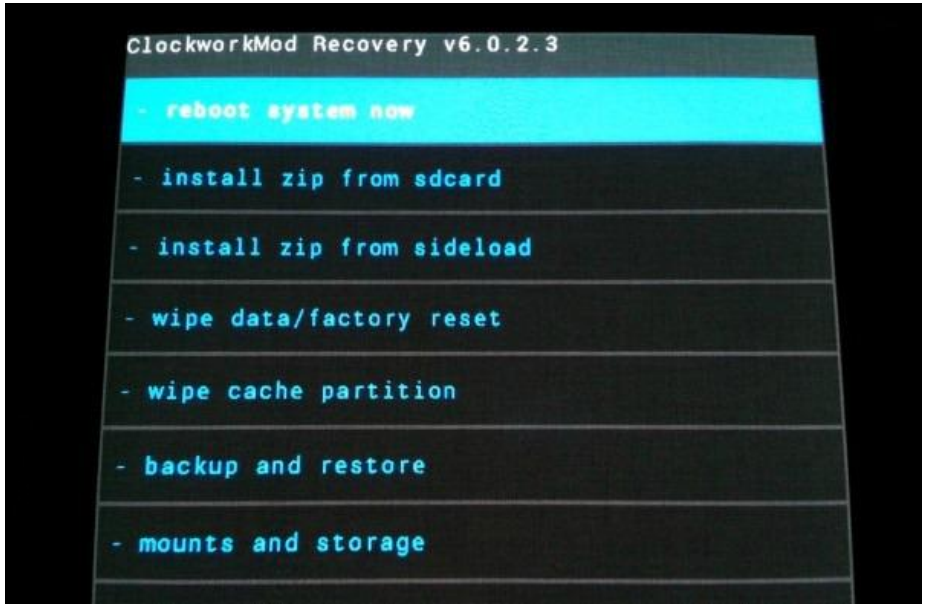
- Started from bootloader
  - Or `adb reboot recovery`
- The capabilities depend on the firmware
  - Which depends on the manufacturer
- Locked
  - However, during flashing official images, it can be overwritten
  - Manufacturer decides what images can be flashed
    - usually third party is not allowed officially
      - It is being checked
- It can be changed via unlocked bootloader
  - Arbitrary bootloader can be installed
  - Then arbitrary main firmware can be installed
  - Cooked ROMs

Recovery

Android „official”



Custom



Reasons to custom recovery

- Contra
  - Loosing the warranty
  - You can brick the phone easily
  - You loose all custom data
  - Arbitrary ROM can be installed
- Pro
  - More freedom
  - Extra functions
    - Backup
    - Restore
  - Arbitrary ROM can be installed

Demonstration – Hands on!

- Device in bootloader mode
  - Nexus 9
  - Pixel 4 XL
  - HTC Sensation
- Investigate its capabilities
  - Use fastboot

Flashing

From this point you have to take full responsibility!

Flashing on Pixel – manual way

- Firmware image from
  - https://developers.google.com/android/images
    - Pixel 4 XL „coral" 10.0.0 (QQ2A.200501.001.B2, May 2020)
    - Pixel 3 XL „crosshatch" 10.0.0 (QQ2A.200501.001.B2, May 2020)
  - Double check it!
  - Reboot in bootloader mode
    - Power + Volume down
    - `adb reboot recovery`
  - Unlock the bootloader
    - `fastboot flashing unlock`
    - Wipes user data!
  - Flash the image
  - Lock the bootloader
    - `fastboot flashing lock`

Flashing on Pixel – OTA

- Firmware image from
  - https://developers.google.com/android/ota
    - Pixel 4 XL „coral" 10.0.0 (QQ2A.200501.001.B2, May 2020)
    - Pixel 3 XL „crosshatch" 10.0.0 (QQ2A.200501.001.B2, May 2020)
  - Double check it!
  - Start in recovery mode
    - Power + Volume up
    - `adb reboot recovery`
  - Upload the zip
    - `adb sideload <filename>`

Flashing on Pixel – Android flash tool

- Android flash tool for Pixel devices
  - https://flash.android.com/welcome?continue=%2Fcustom
  - Do as it says.

To root or not to root?

- What does it mean?
  - You can execute commands with root (unix) permissions
    - Even from shell
  - The `\system` is mounted in read-write mode
    - You can install busybox as well
- How?
  - Bootloader have to be unlocked first
  - Custom recovery and Custom system is not required
    - However in most of the cases rooted phones runs custom systems
- Why?
  - Clean up the pre-installed applications
  - Change system level settings
    - Overclock
    - Energy management
    - USB modes
    - …

Homework

- Project final submission!
  - Deadline 5/20 midnight

The End