Use Android Generic System Images (GSI)
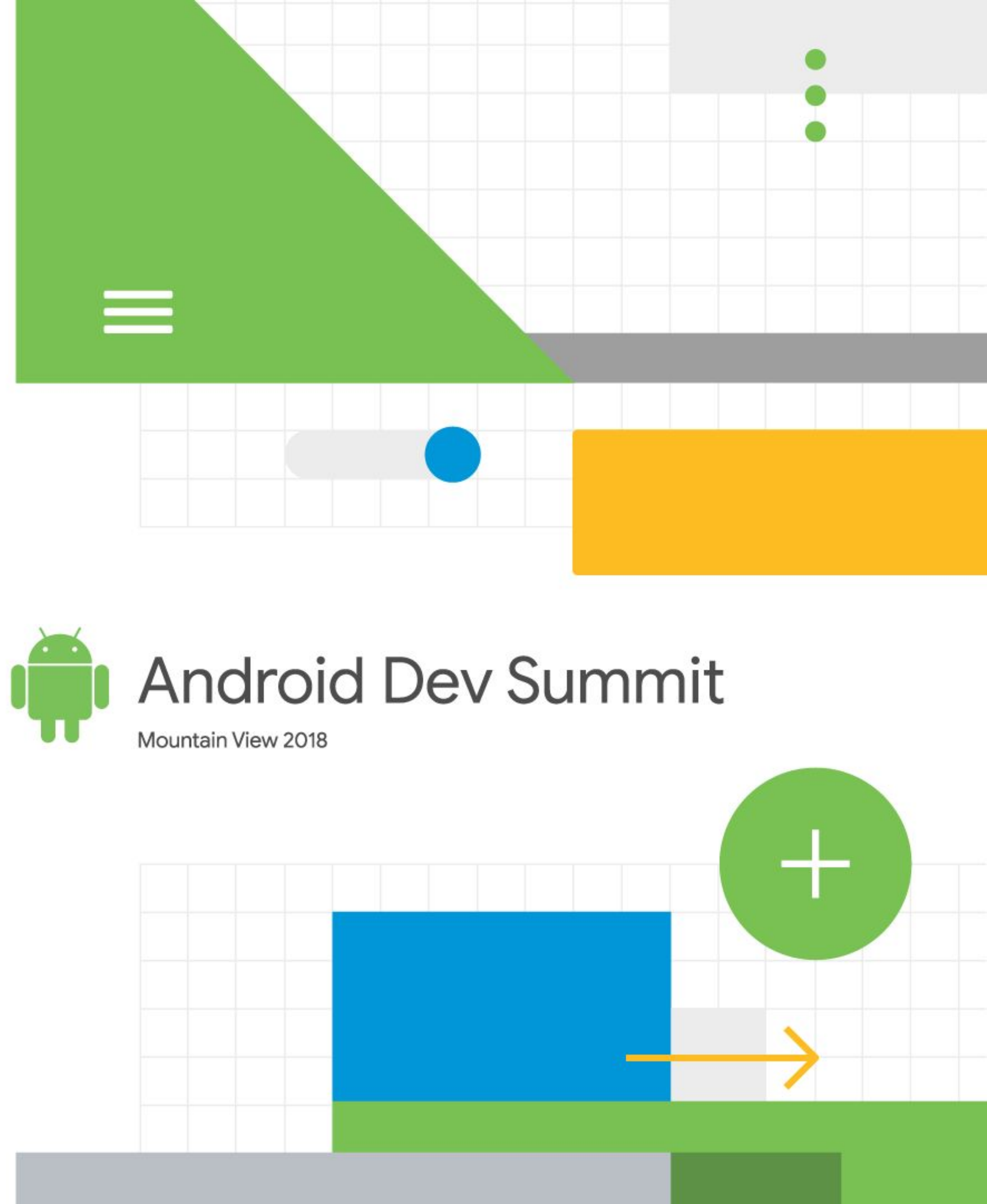
A How-To Approach

Gfan@google.com
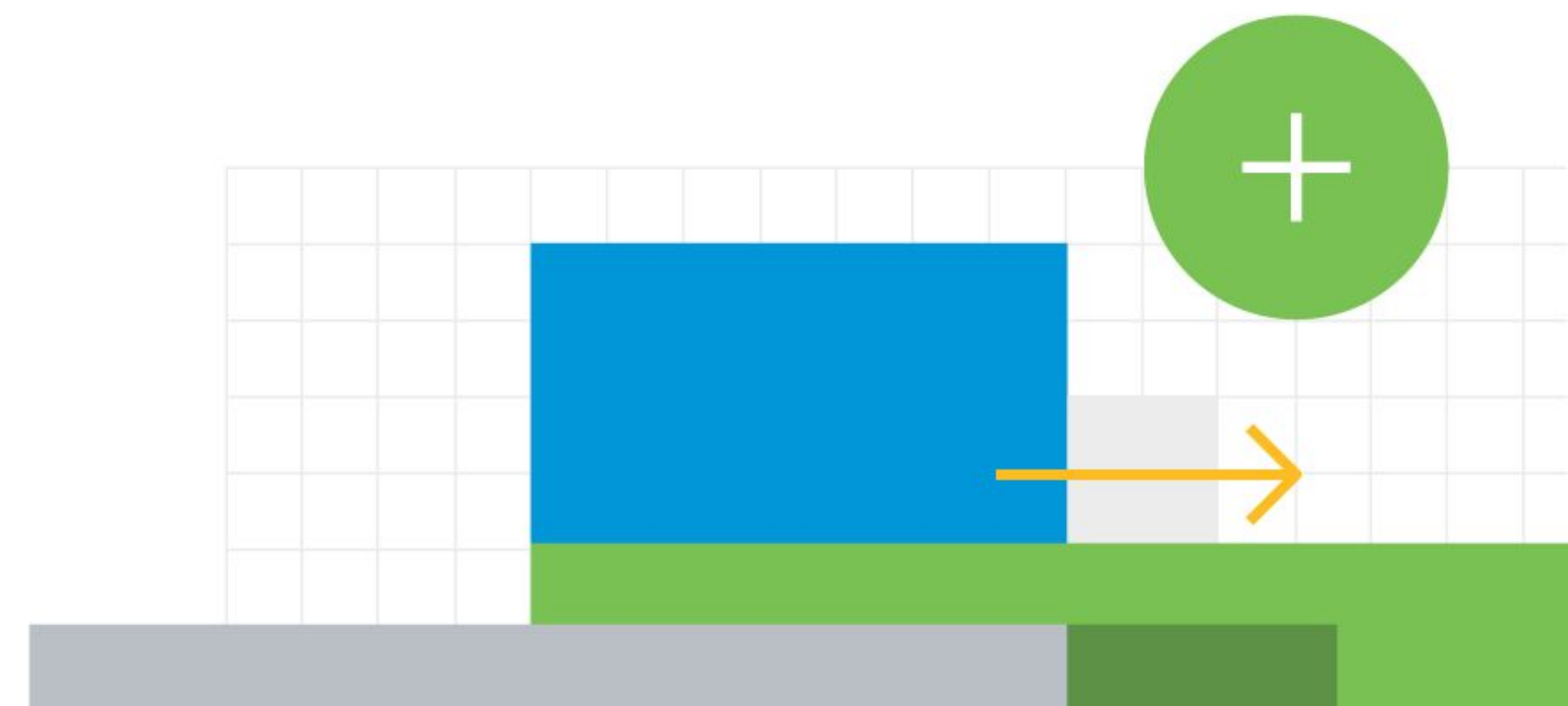
Android Dev Summit

Mountain View 2018

GSI @ 1000ft
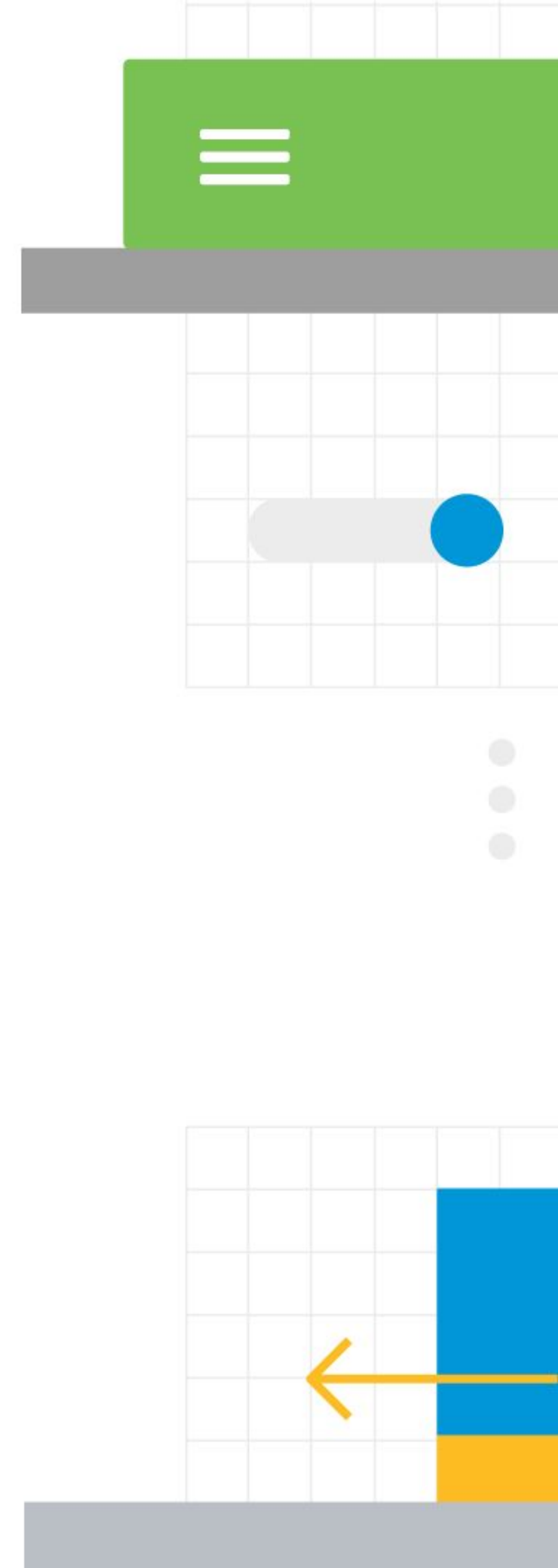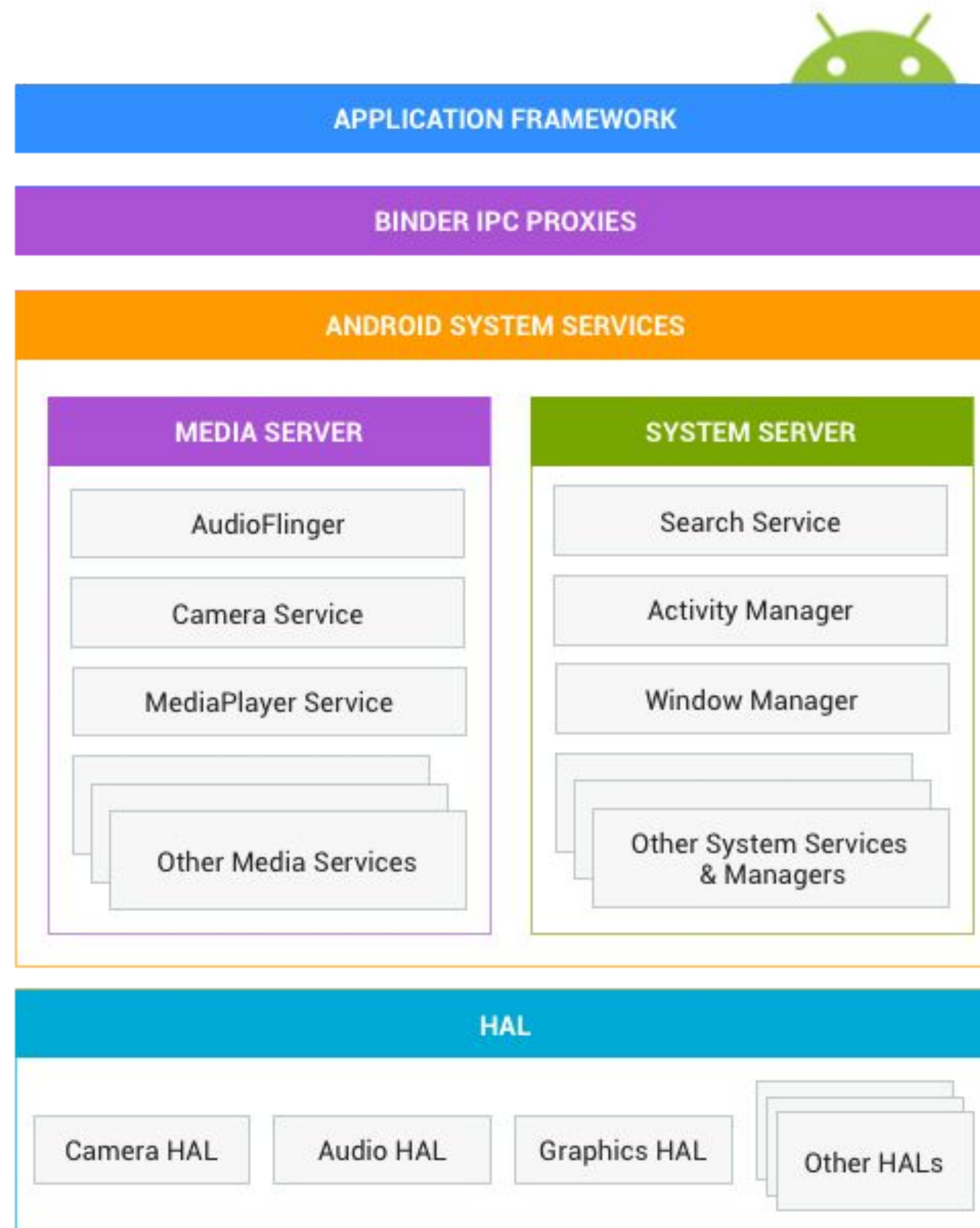
Generic system images (GSI) is Useful!

- Develop app with new OS features
- Validate app for future OS compatibility
- Report OS issues conflicting with your app & get them fixed by Android Team!
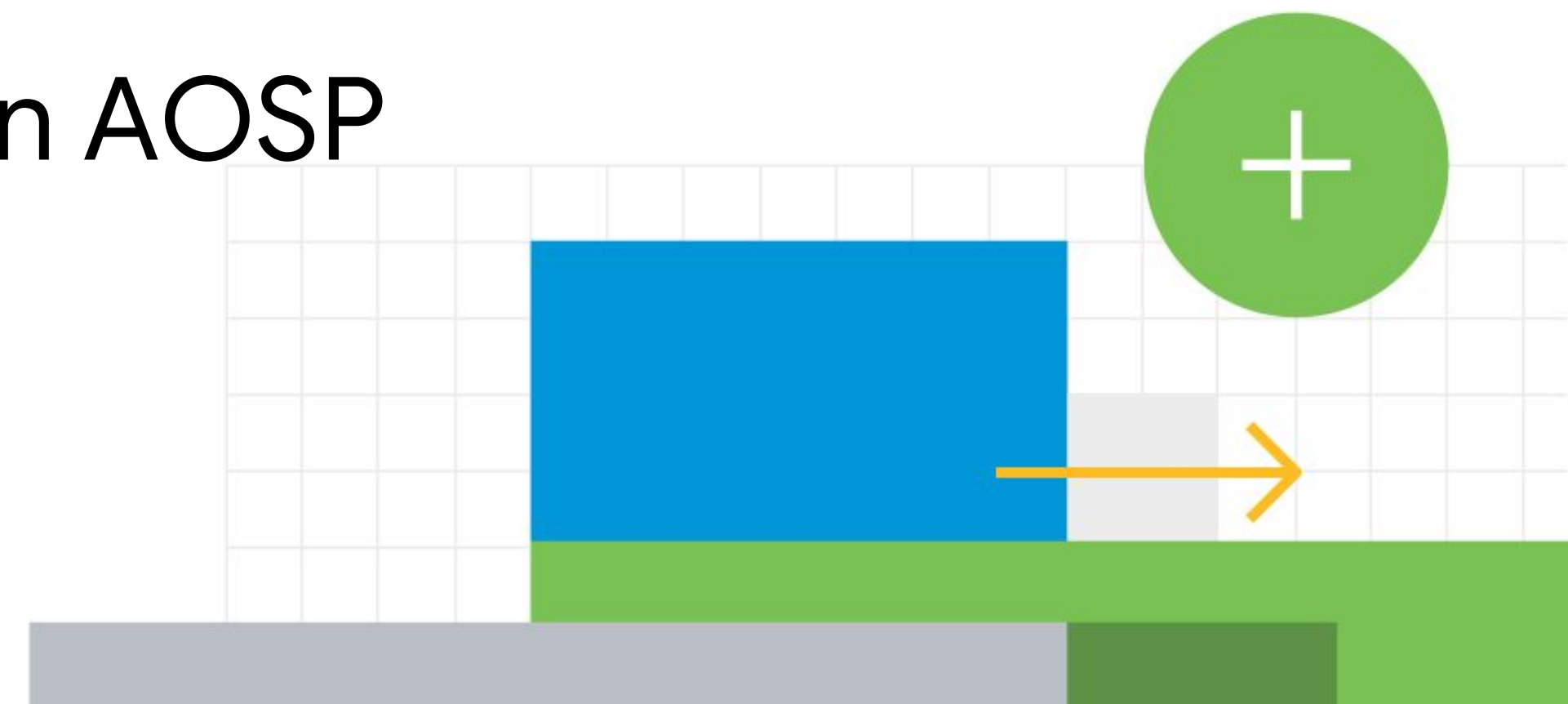
GSI Name

Physically GSI is a file and got a name too:

system.img

Where is it?

We build it: the only way to get it for now!

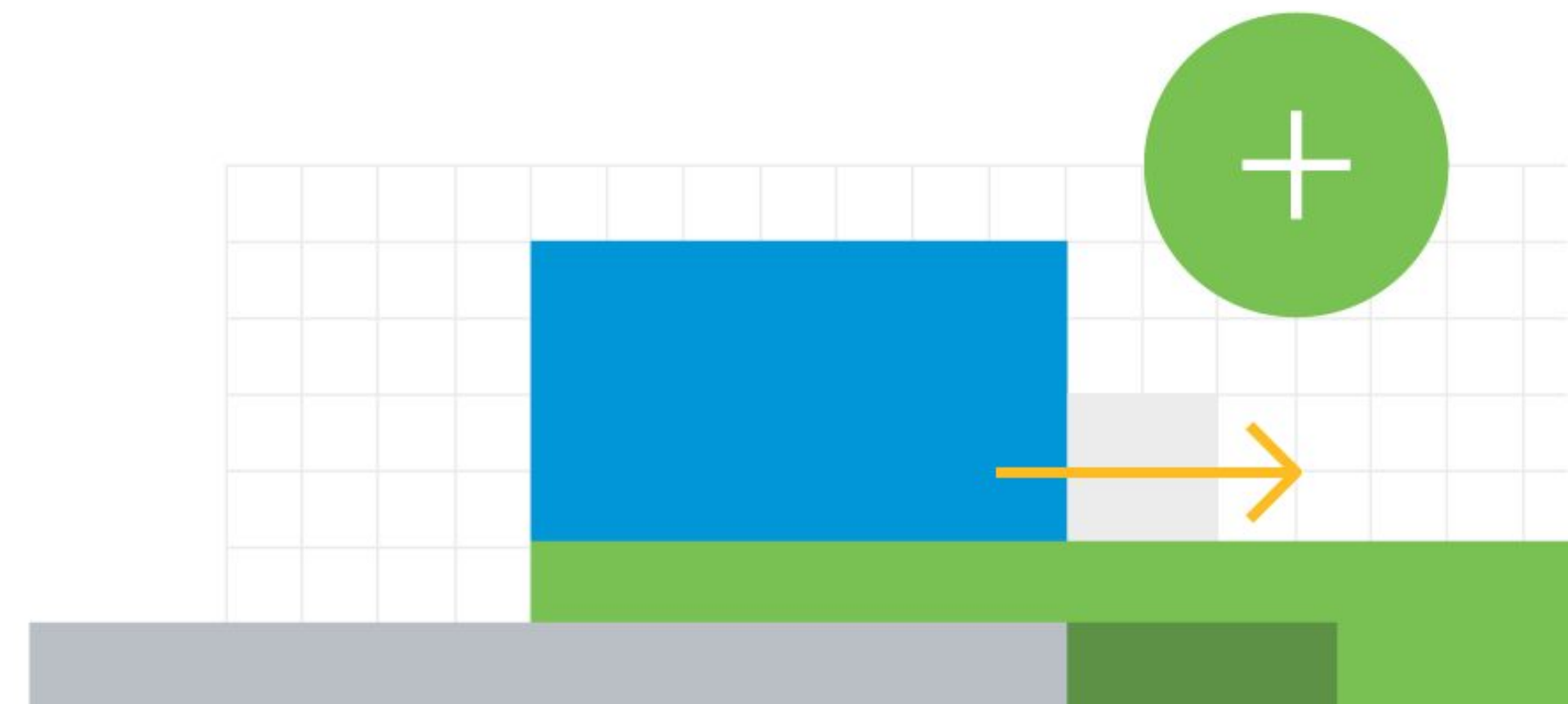The code is what we love! Code is on AOSP

Building GSI

What needed to build:

- Branch name
- Lunch menu

let us get them!
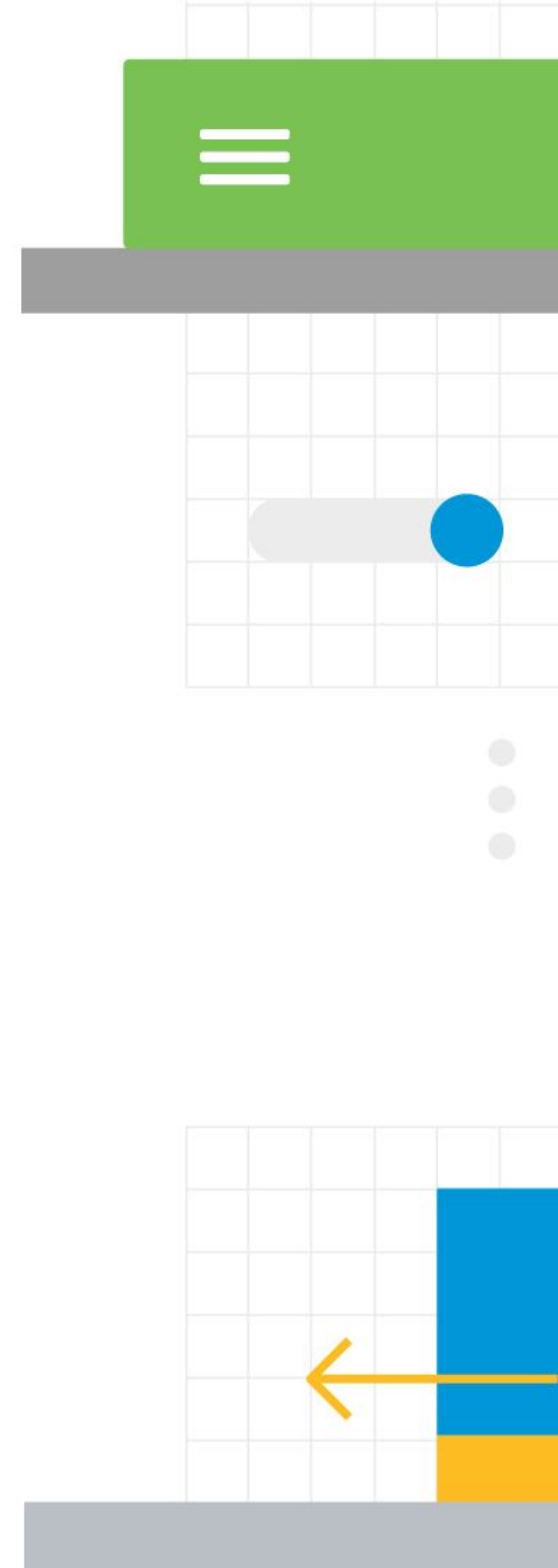
GSI compatibility

- Requires Treble Support

```
if( Treble Supported ) {
    if ( Fully Treble Compliant ) {
        Use the same OS ver or newer of GSI;
    } else  {
        Use GSI version = On Dev OS ver;
}
```

- Treble Compliance Landscape
  - New Android 9 devices: full
  - Upgraded Android 9 dev: full, partial, or not

GSI compatibility

- ## Treble Check
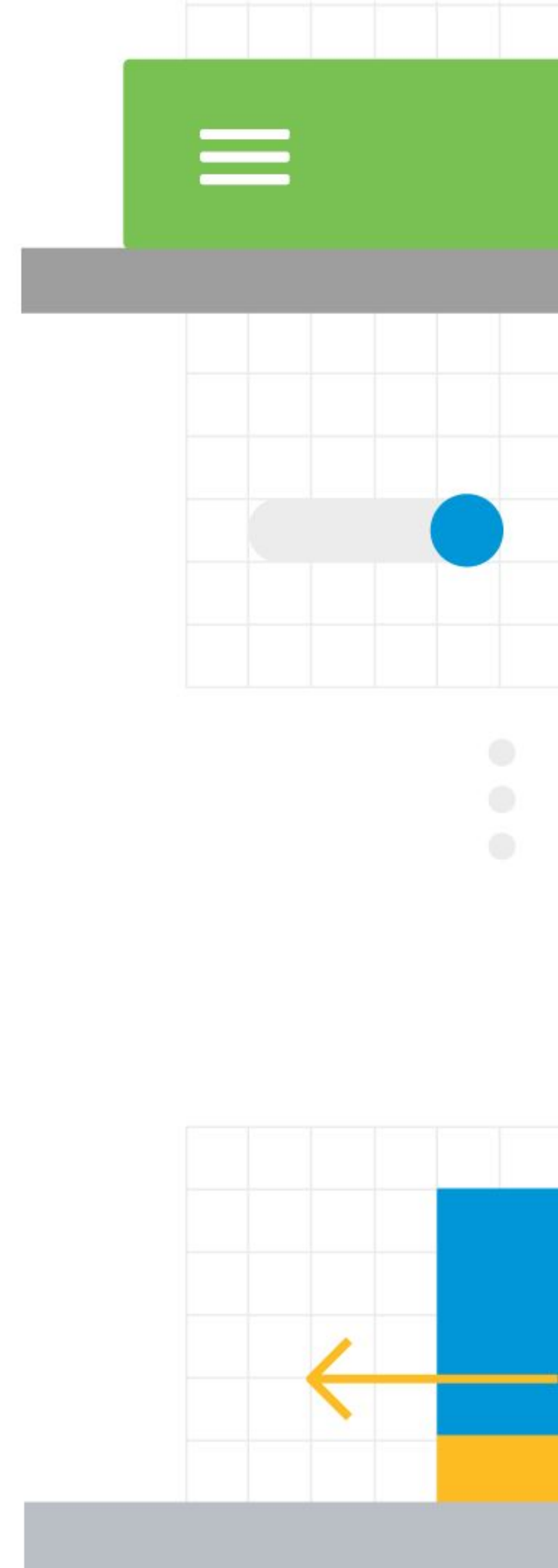
  ```
  $adb shell getprop ro.treble.enabled
  ```

- ## Full Treble Support

  ```
  $adb shell cat /system/etc/ld.config.txt
  ...
  [vendor]
  namespace.default.isolated = true
  ```
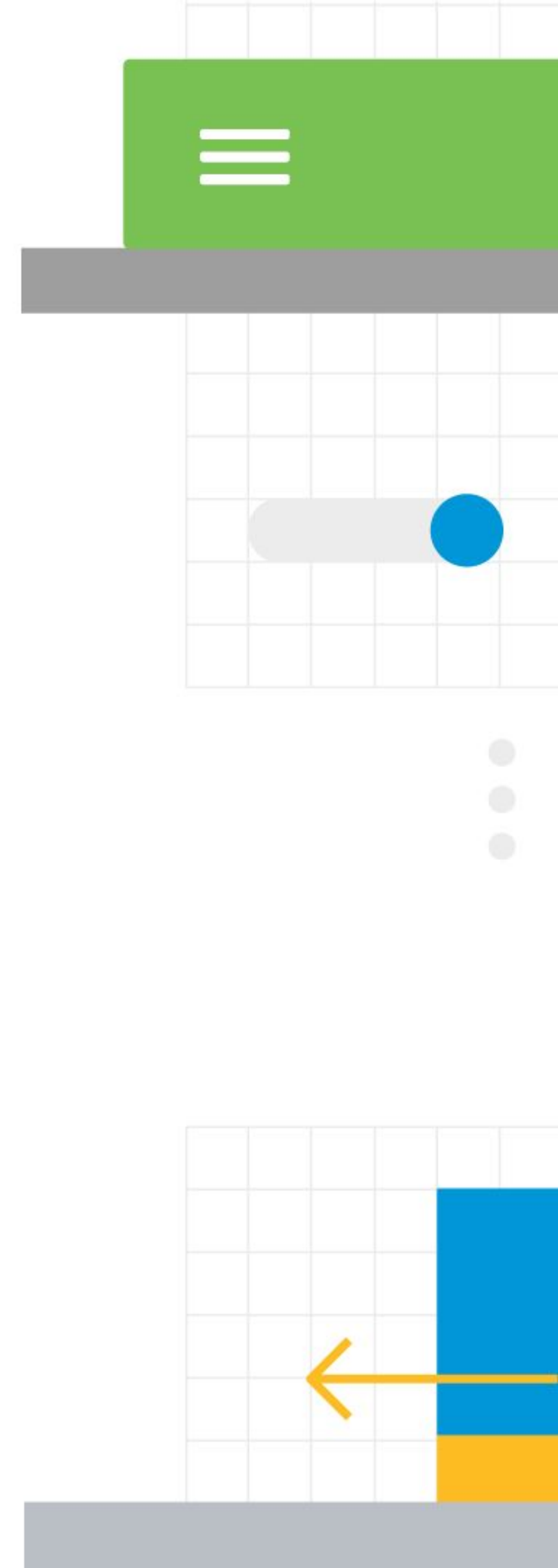
Select GSI: $cpuArch

- Use the same CPU architecture as the device

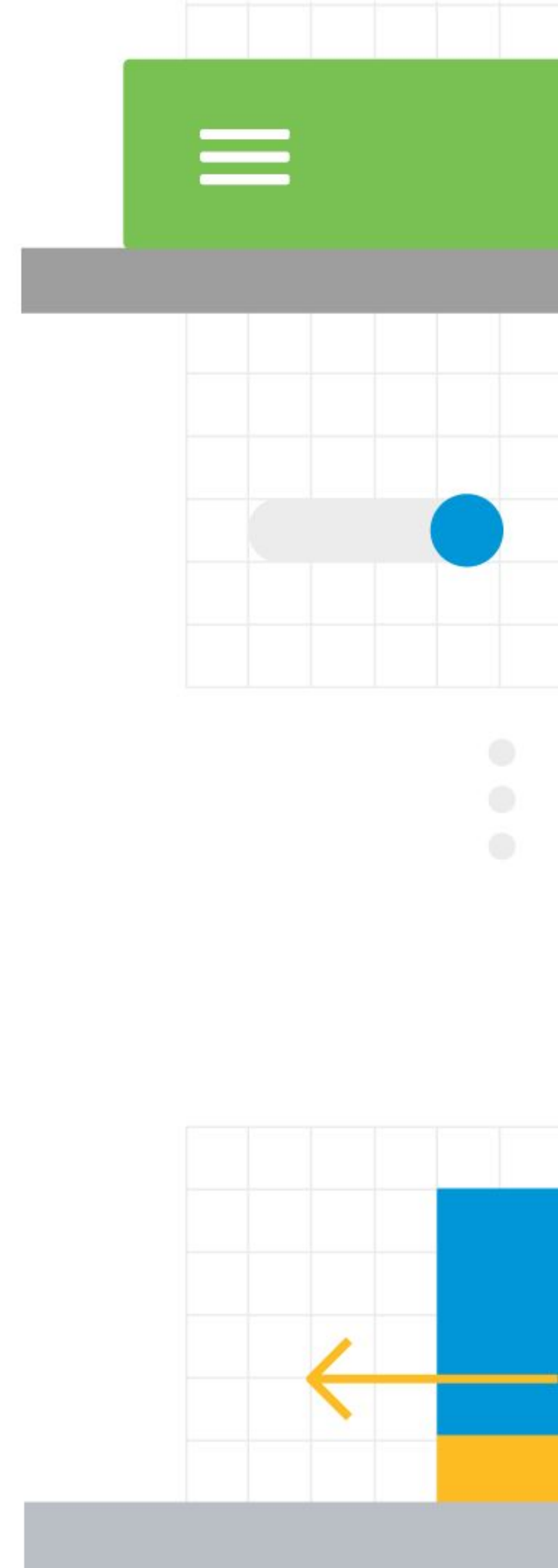```
$adb shell getprop ro.product.cpu.abi
arm64-v8a
```

# Select GSI: $system-privilege

- Use the same privilege as the device

```
$adb shell cat /proc/mounts | grep -q
/dev/root && echo "system-as-root" || echo
"non-system-as-root"

system-as-root
```
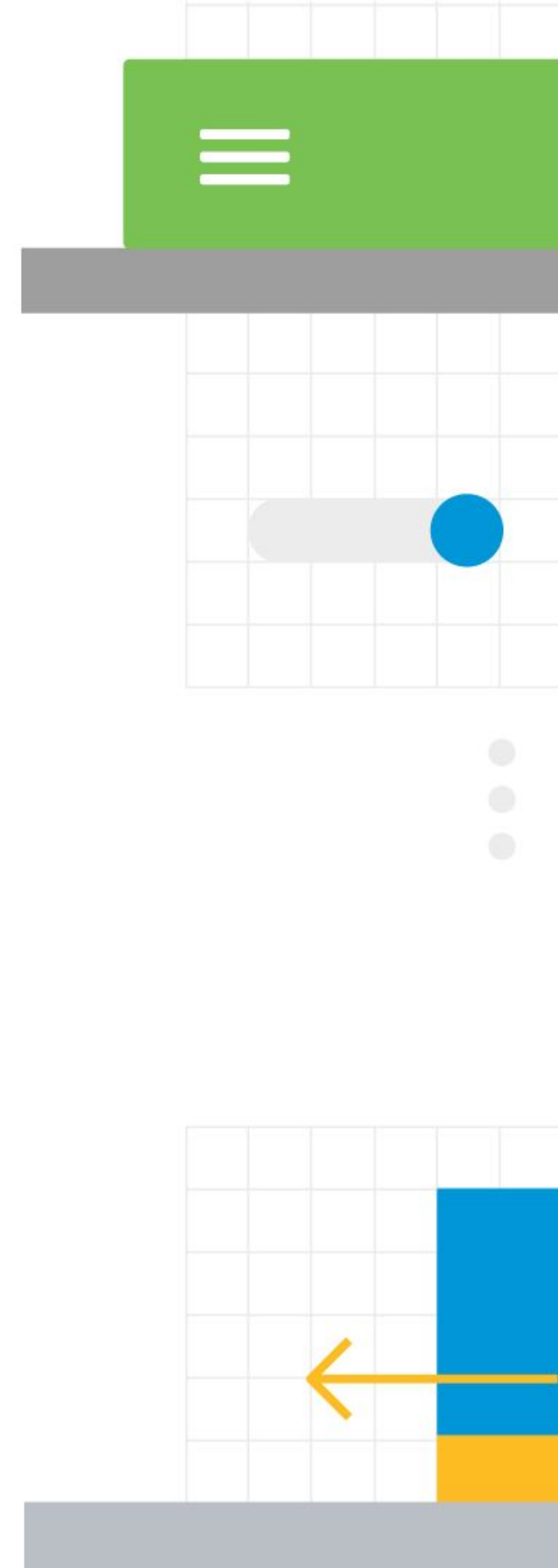
- Determine the flavor
  - system-as-root: *$cpuArch_ab*
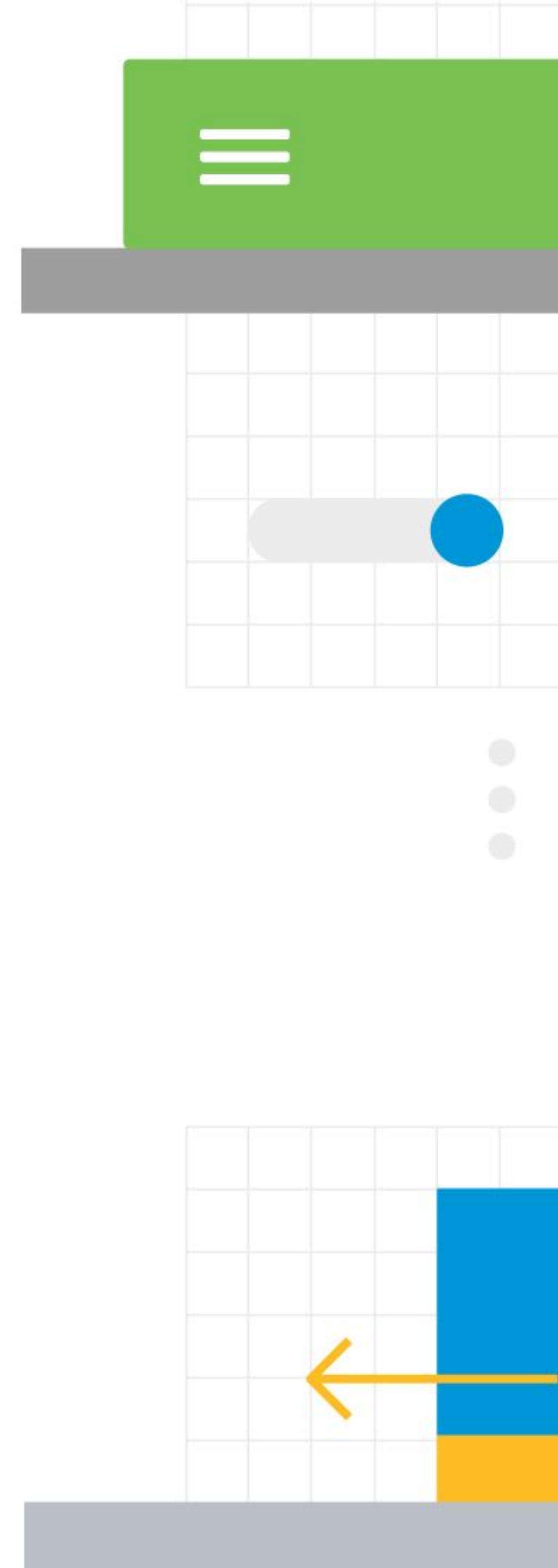  - non-system-as-root: *$cpuArch_a*

# Select GSI: summary

- Got the right GSI flavors
  - OS ver => Android 9
  - $cpuArch  => arm64
  - System Priv => _ab

- GSI availability
  - Open Source in AOSP
  - *Pre-build Image in the future*

So we build GSI!

# Build GSI: requirements

- Linux Platform
- Download Source
- Right *menu* for *lunch*
  - use result from previous steps

# Build GSI: let's do it

- Get Android [Source code](#):

```
$repo init -u
https://android.googlesource.com/platform/manif
est -b pie-gsi

$repo sync -j8

(takes about half day to finish… because: it is
10 years work)
```
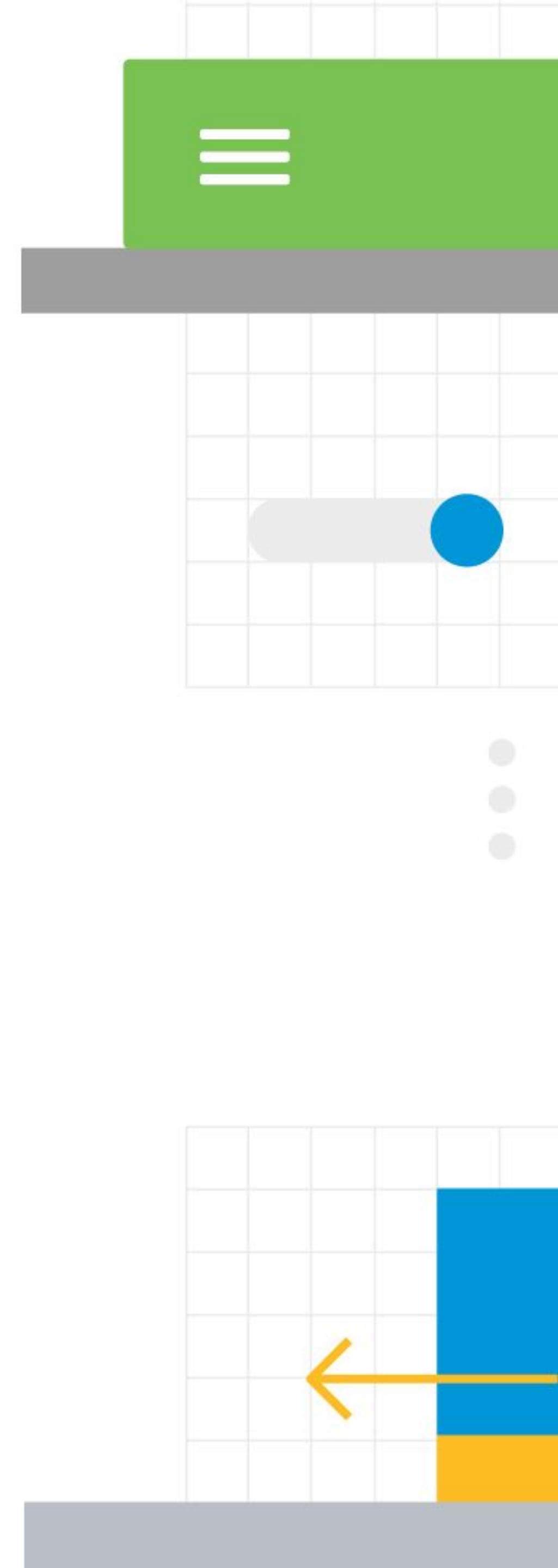
- Set up build environment

```
$source build/envsetup.sh

$lunch   aosp_arm64_ab-userdebug
```
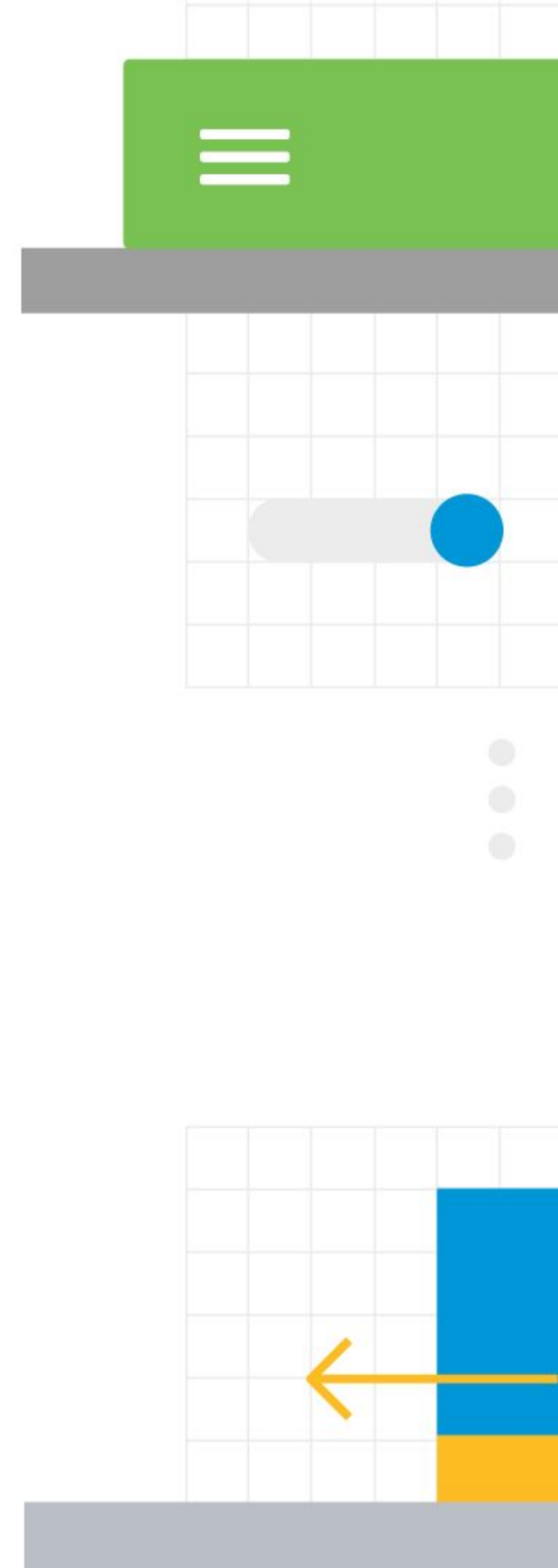
# Build GSI: really building it
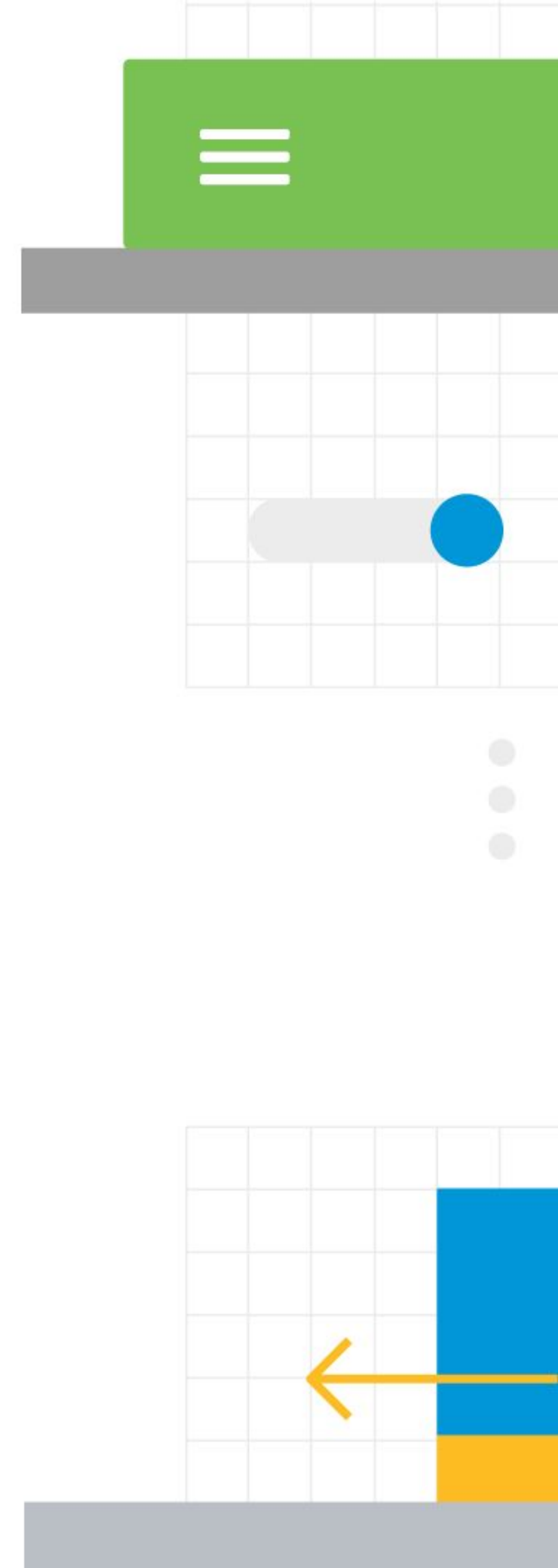
- Simply do

  ```
  $make -j
  ```

- Check out the result

  ```
  $ls -s out/target/product/generic_arm64_ab/system.img
  1259172 system.img
  ```

Install GSI

- Facts
  - Unlockable Treble devices
  - Steps for Unlocking & installation
    - *manufacturer dependent*
    - *fastboot or variation of it*

- Generic Steps
  - Unlock
  - Disable verify boot
  - flash

Install GSI: a Pixel example
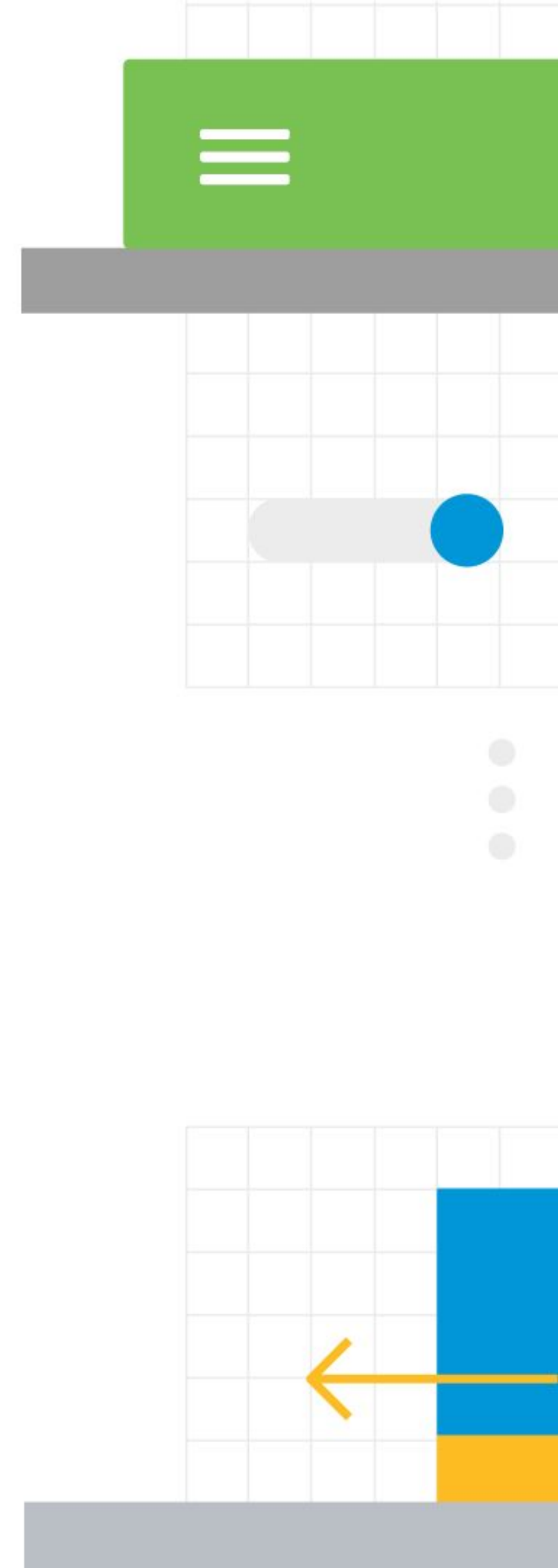
- ● Unlock Pixel

  ```
  $fastboot flashing unlock
  ```
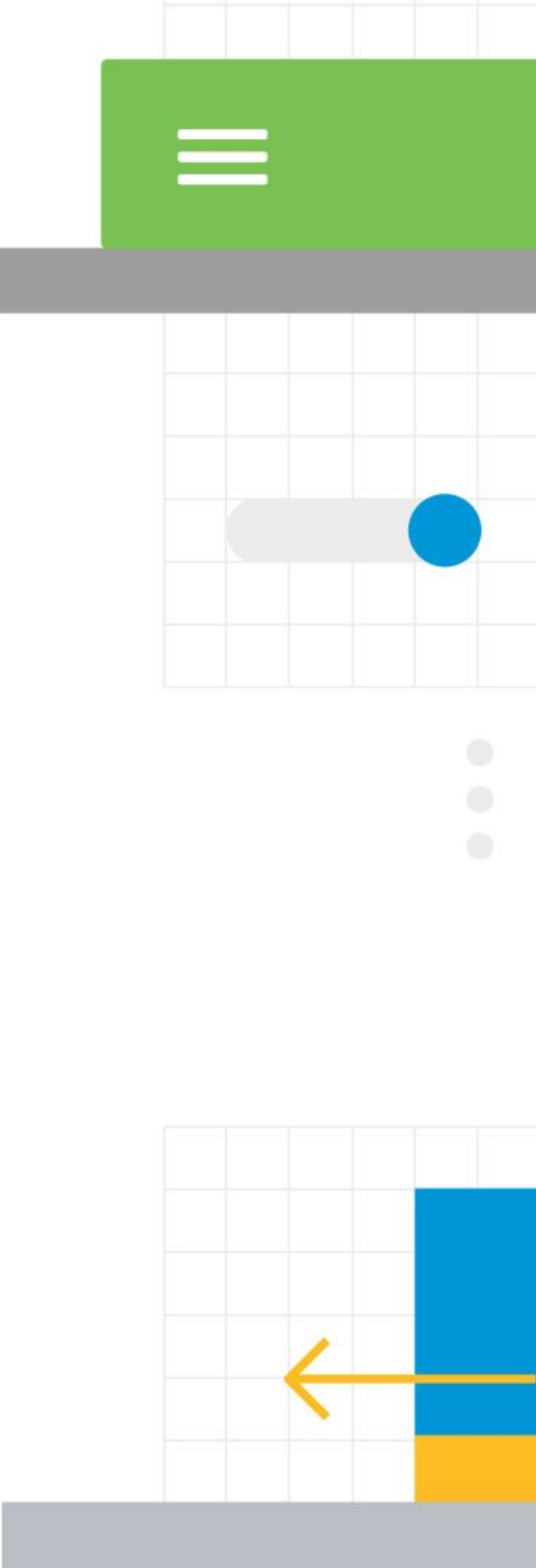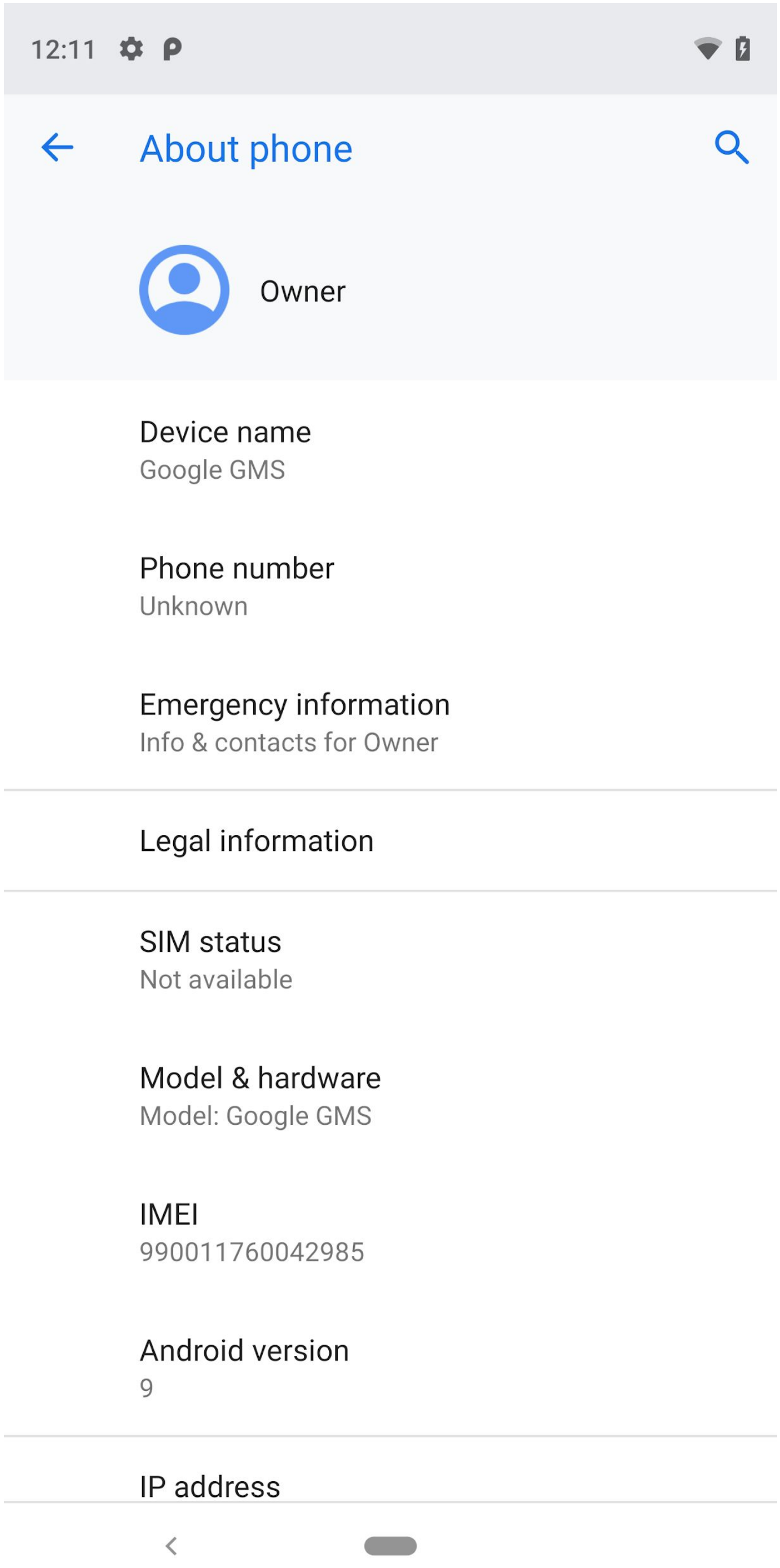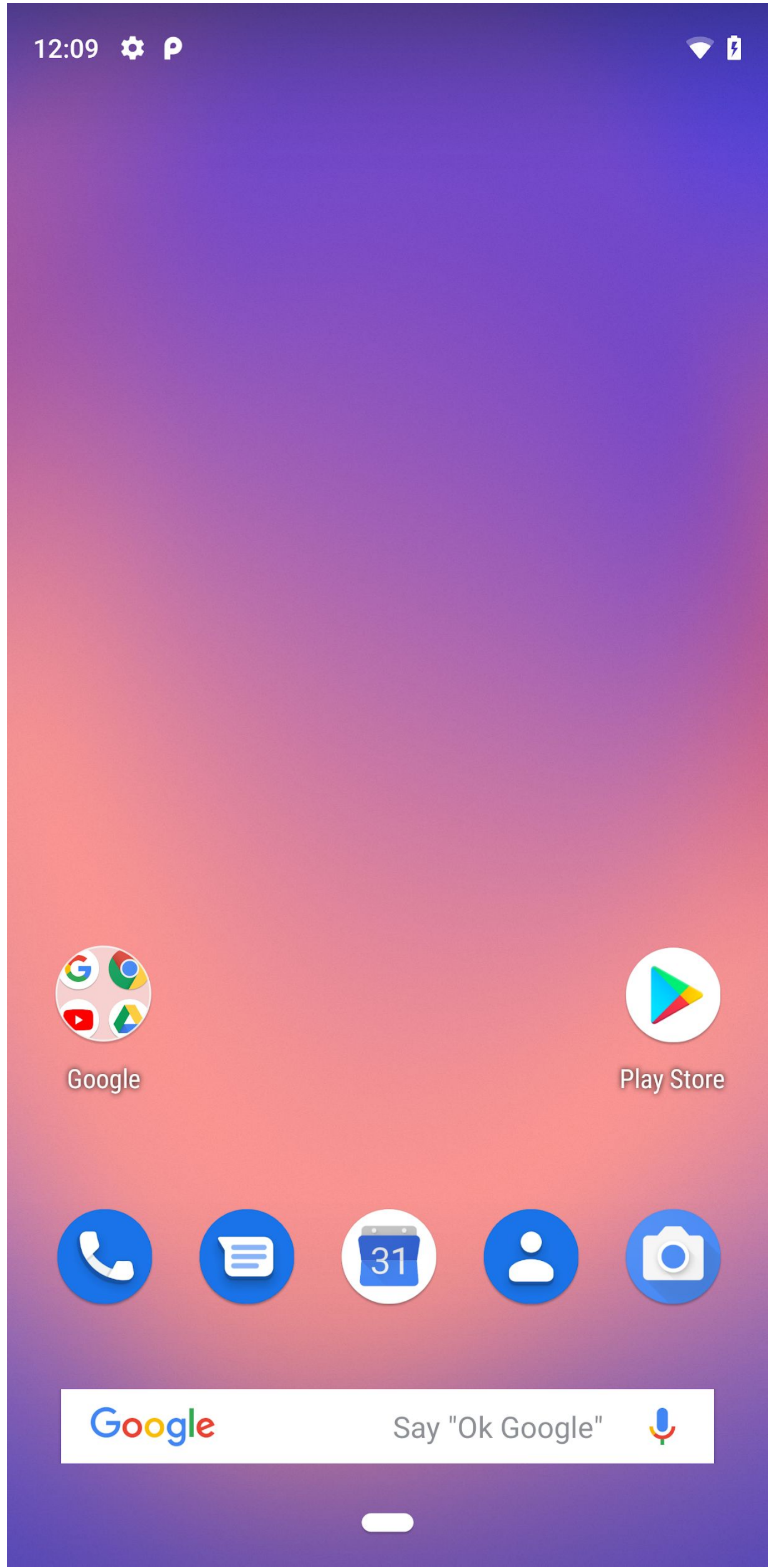
- ● Disable Verity Boot

  ```
  $fastboot flash vbmeta
  out/target/product/generic_arm64_ab/vbmeta.img
  ```

- ● Install GSI, really doing it

  ```
  $fastboot erase system

  $cd out/target/product/generic_arm64_ab

  $fastboot flash system system.img

  $fastboot reboot -w
  ```

Google

Play Store

Google | Say "Ok Google" 🎤

← **About phone** 🔍

Owner

Device name
Google GMS

Phone number
Unknown

Emergency information
Info & contacts for Owner

Legal information

SIM status
Not available

Model & hardware
Model: Google GMS

IMEI
990011760042985

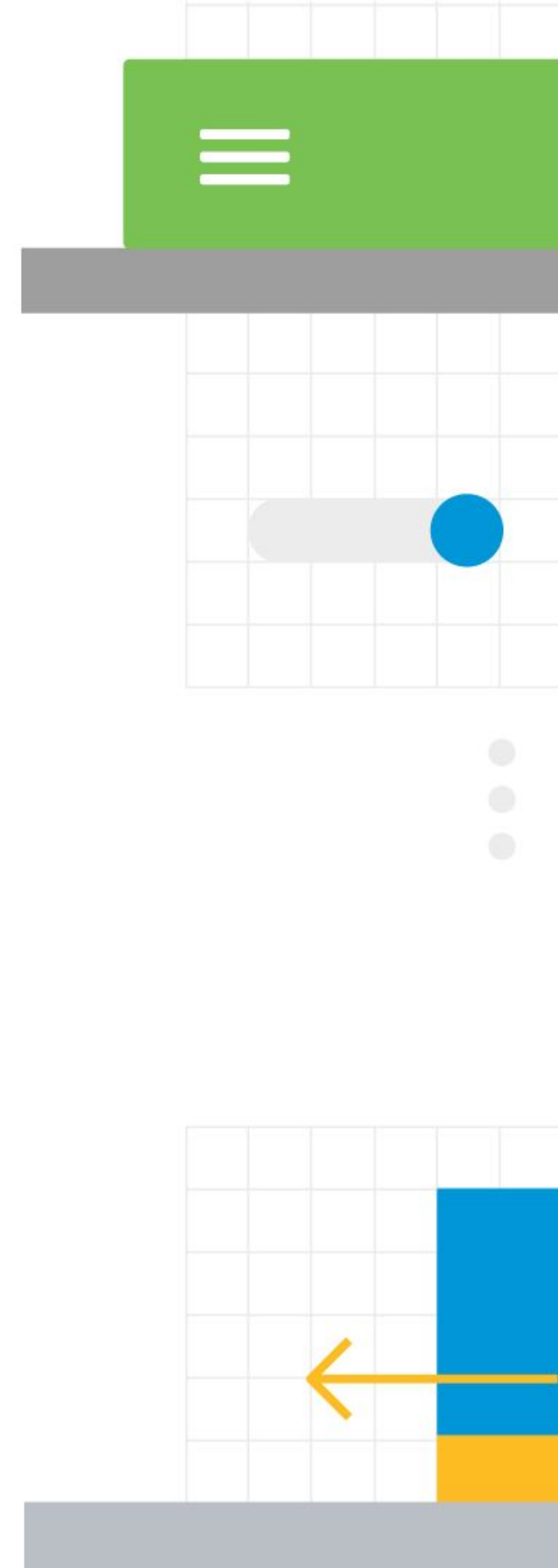**Android version**
9

IP address
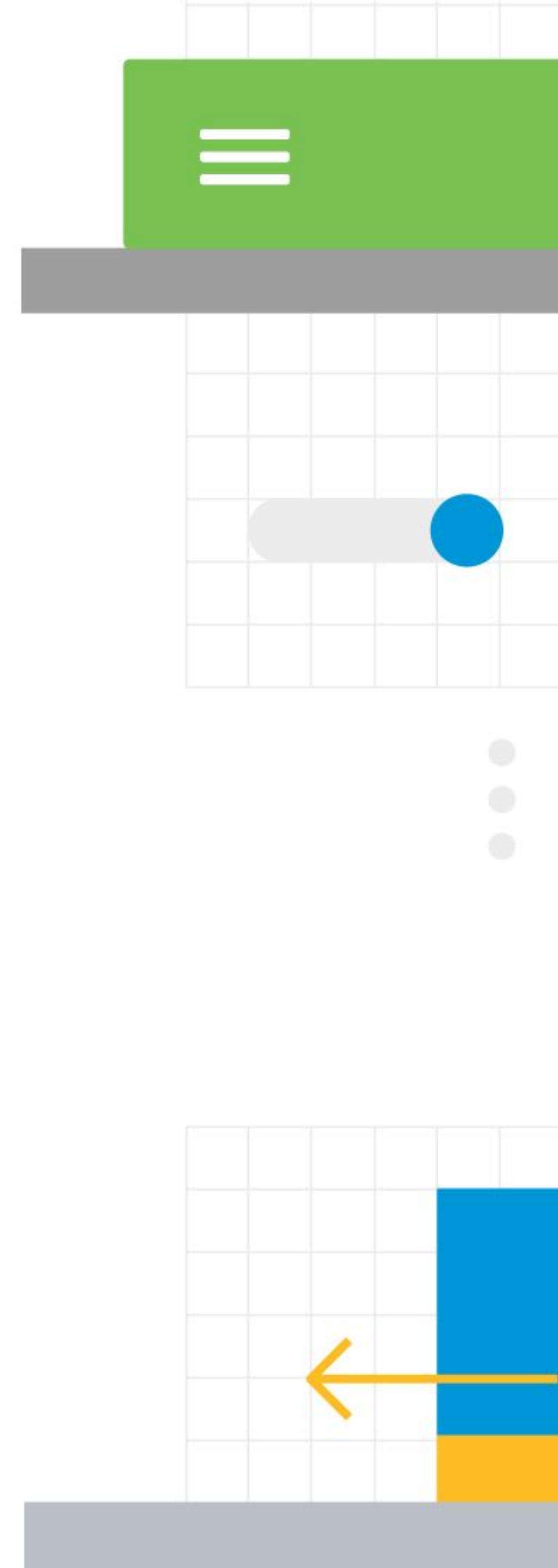
Let GSI work for you

- Develop apps with new OS features

- Validate existing apps for future compatibility

- Improve OS quality

GSI enables you to get hands on new OS earlier!

Next Steps

- Use GSI to your advantage

- Create PR on [AOSP](#) for GSI

- Report [GSI issues](#)

Other Materials

- Android Pie GSI source code

- GSI documentation
  - on developer.android.com
  - on AOSP

- GSI Community
  - XDA developers
  - Stack Overflow