

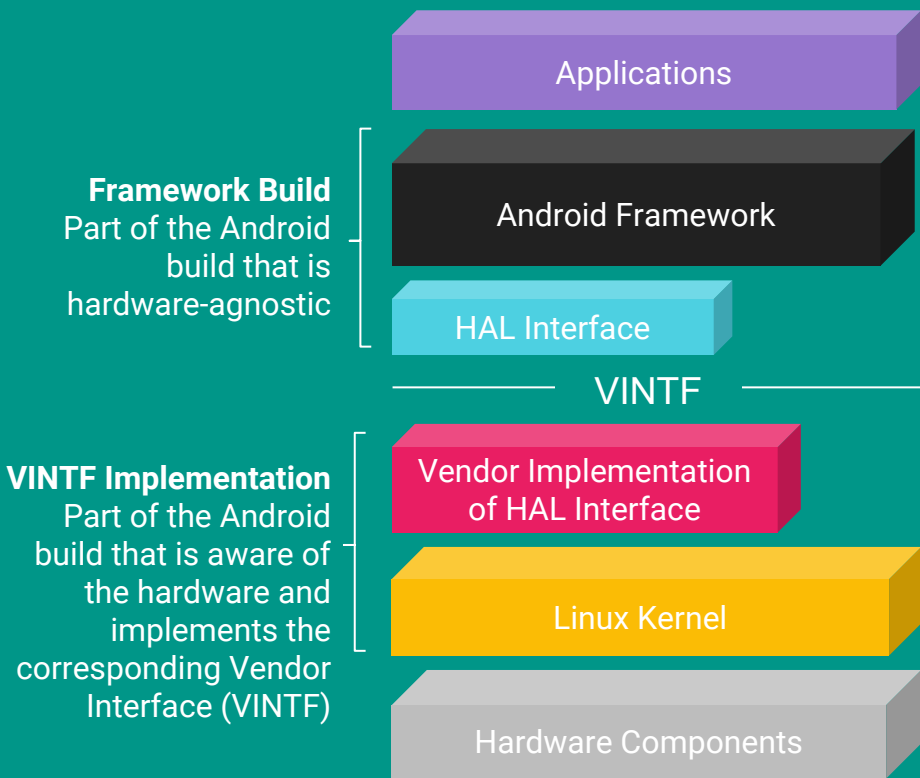
Project Treble and Kernel

Linux Plumbers, September, 2017

Sandeep Patil <sspatil@google.com>

Project Treble & kernel

- Vendor Interface (VINTF)
 - Collection of versioned HAL interfaces
 - Kernel interfaces exposed to userspace
- Vendor Test Suite (VTS)
 - Collection of tests to verify implementation of VINTF
 - VTS tests ensure a VINTF implementation works well with a future Android framework
- Linux Kernel
 - Foundation of Android platform
 - Framework directly depends on core kernel interfaces
 - Target for VTS testing



Android Kernel requirements (before Oreo)

“You can use any version of the kernel as long as it supports the required features (such as the binder driver). However, we recommend using the latest version of the Android kernel.”

Problems

Older kernels

Most Android devices ship with Linux kernels released between 2 - 5 years before the device is shipped.

Multiple kernel versions

Android devices ship with a variety of kernel versions (3.4.y, 3.10.y, 3.18.y, etc). Each core kernel patch or security fix gets applied differently across the ecosystem.

Slower patch propagation & kernel updates

Time taken for kernel changes to propagate from android-common kernel to the device is directly proportional to the number of kernel trees the patch flows through.

Project Treble goals

Kernel updates

- Faster device kernel updates for bug/vulnerability fixes.
- Easier minor version bumps for device kernel.
- Add consistency in kernel maintenance across the Android ecosystem.

Separation of deliverables in kernel

Kernel running on a device is a mix of:

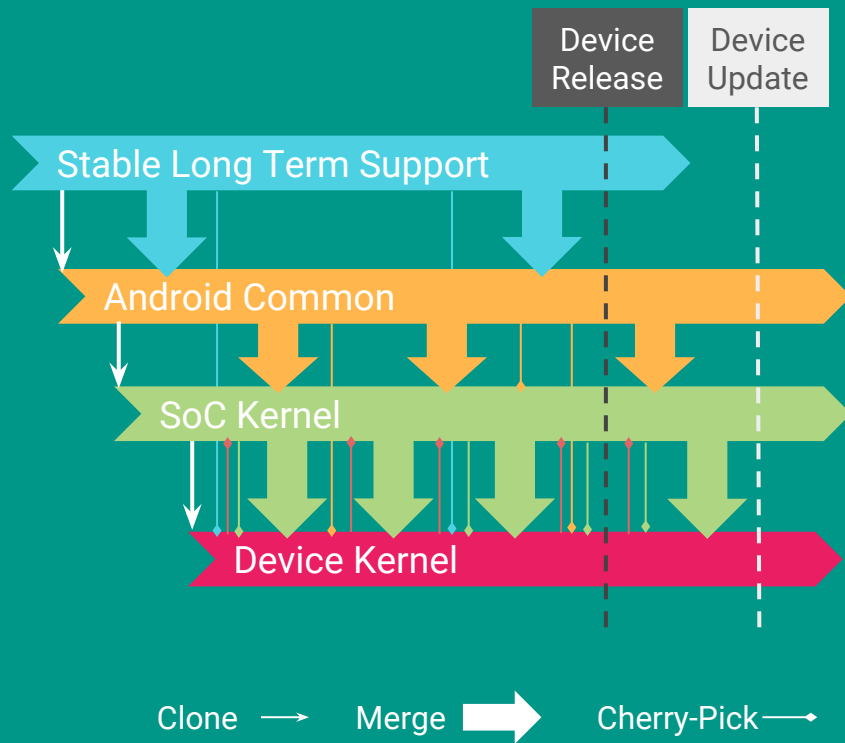
- Long Term Stable
- Android Common
- SoC Kernel
- ODM Changes

Ability to test kernel ABI/APIs to ensure Android compatibility

Android framework directly depends on several kernel interfaces. As part of Treble, we want to specify those and also have tests against them to ensure future Android releases continue to work with a given kernel.

Device kernels

- Patches/Backports/Features sourced from different trees at random intervals
- No Separation of SoC/ODM code
- Every device kernel using same SoC is different
- *Make downstream cherry-picks obsolete*



Treble additions to Android

- Adding support for mounting `vendor` and `system` partitions early during boot (First stage mount)
- Adding new module utilities (modprobe)
- Build system changes for installing modules
- Tool to generate Device tree overlay partition image

Minimum Kernel Version

- All SoCs productized in 2017 to use kernel v4.4 or newer.
- All other SoCs launched in Android 8.0 devices to use kernel v3.18 or newer.

SoC <-> Manufacturer separation

- Use Kernel modules for board peripherals instead of building them into the kernel.
 - CONFIG_MODVERSIONS for module versioning
- Use Device tree overlays to describe the board and the kernel device tree to describe the SoC
- Overlay to be maintained in a separate image / partition
- Bootloader merges base device tree with the board overlay based on hardware identifier
 - No runtime overlays
- /system and /vendor are mounted in Init's first stage.
 - Allows for kernel modules to be inserted very early during boot
 - Allows for the separation of SELinux policy
- Track and test kernel dependencies in VTS
- Existing Kernel tests in VTS : LTP, kselftest and other tests.

Kernel dependencies

- Support for `/proc/config.gz`
- All kernel configurations listed in [android-base.cfg](#)
- All `procfs`, `sysfs` entries that are directly relied upon by Android. E.g. `/proc/meminfo`, `/proc/kmsg`, `/sys/power/wake_lock` etc.
- List: <https://source.android.com/devices/architecture/kernel/reqs-interfaces>
- More

THANK YOU