



Das Android Open Source Project

Oder: “Alles, was du schon immer über die Software auf deinem Android Smartphone wissen wolltest.”



Stefan Lengfeld

Application Development

Android & Linux Embedded Entwickler

2014-2016: Linux Entwicklung in Mainz

Seit 2017: Linux & Android Entwicklung
bei inovex in Köln

Um was soll es heute gehen?

- › 1. Teil:
 - › Geschichte von Android
 - › Hardware
 - › Betriebssystem-Komponenten
 - › Sicherheit
 - › “Project Treble”
- › 2. Teil:
 - › Development Setup
 - › Download, building and flashing of the AOSP

1. Teil – Theorie

Was ist dieses Betriebssystem,
das 2 Milliarden User verwenden?

Quelle: [https://en.wikipedia.org/wiki/Android_\(operating_system\)](https://en.wikipedia.org/wiki/Android_(operating_system))

Android ist ein Ökosystem und
Trademark von Google.

Das AOSP ist ein Smartphone
Betriebssystem.

Geschichte von Android

- › “Android Inc. was founded in Palo Alto, California, in October 2003 Andy Rubin, [...]”
- › “In July 2005, Google acquired Android Inc. for at least \$50 million”
- › 2007 Open Handset Alliance
- › 2013 Andy Rubin verlässt das Android Team und später Google
- › 2016 Pixel und Pixel XL Smartphones

Code name	Version number	Linux kernel version ^[3]	Initial release date	API level
(No codename) ^[4]	1.0	?	September 23, 2008	1
Petit Four ^[4]	1.1	2.6	February 9, 2009	2
Cupcake	1.5	2.6.27	April 27, 2009	3
Donut ^[5]	1.6	2.6.29	September 15, 2009	4
Eclair ^[6]	2.0 – 2.1	2.6.29	October 26, 2009	5 – 7
Froyo ^[7]	2.2 – 2.2.3	2.6.32	May 20, 2010	8
Gingerbread ^[8]	2.3 – 2.3.7	2.6.35	December 6, 2010	9 – 10
Honeycomb ^[9]	3.0 – 3.2.6	2.6.36	February 22, 2011	11 – 13
Ice Cream Sandwich ^[10]	4.0 – 4.0.4	3.0.1	October 18, 2011	14 – 15
Jelly Bean ^[11]	4.1 – 4.3.1	3.0.31 to 3.4.39	July 9, 2012	16 – 18
KitKat ^[12]	4.4 – 4.4.4	3.10	October 31, 2013	19 – 20
Lollipop ^[13]	5.0 – 5.1.1	3.16	November 12, 2014	21 – 22 ^[14]
Marshmallow ^[15]	6.0 – 6.0.1	3.18	October 5, 2015	23
Nougat ^[16]	7.0 – 7.1.2	4.4	August 22, 2016	24 – 25
Oreo ^[17]	8.0 – 8.1	4.10	August 21, 2017	26 – 27
Pie ^[18]	9.0	4.4.107, 4.9.84, and 4.14.42	August 6, 2018	28

Legend: ■ Old version ■ Older version, still supported ■ **Latest version**

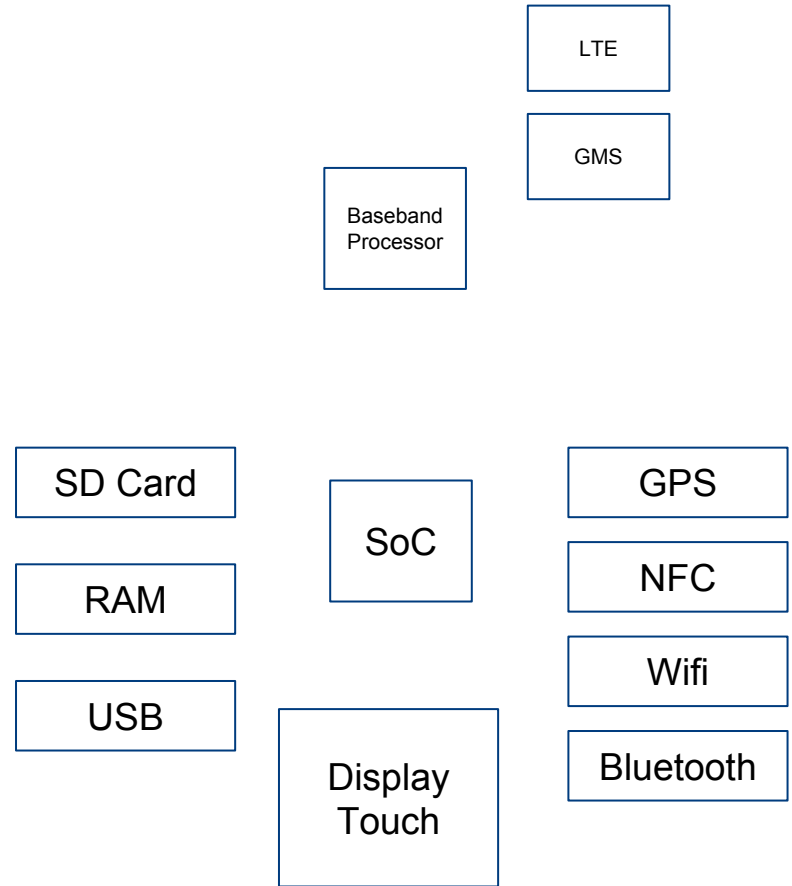
Android Versionen

Quelle: https://en.wikipedia.org/wiki/Android_version_history

Hardware

eines Smartphones

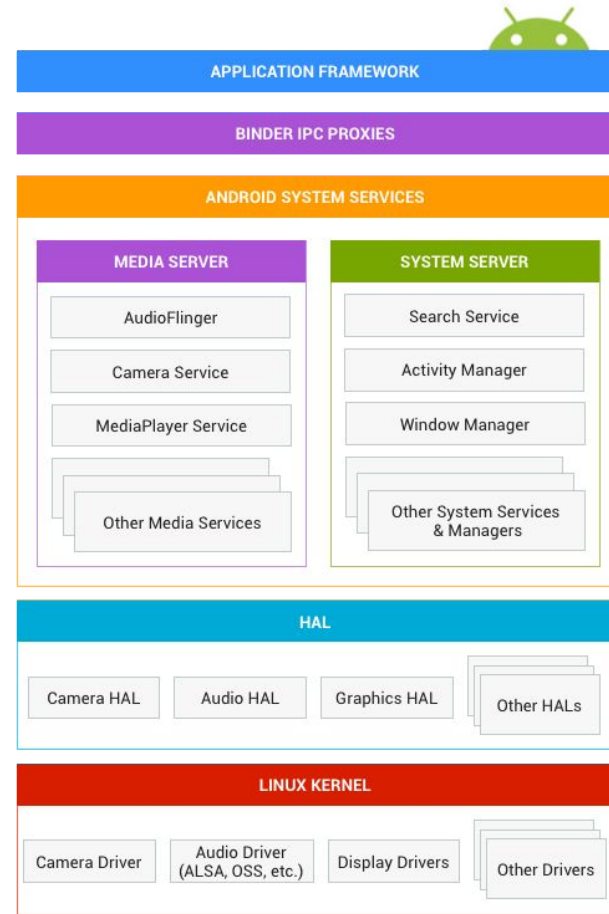
- › System On a Chip (SoC)



OS Komponenten

Linux-basierend, aber ...

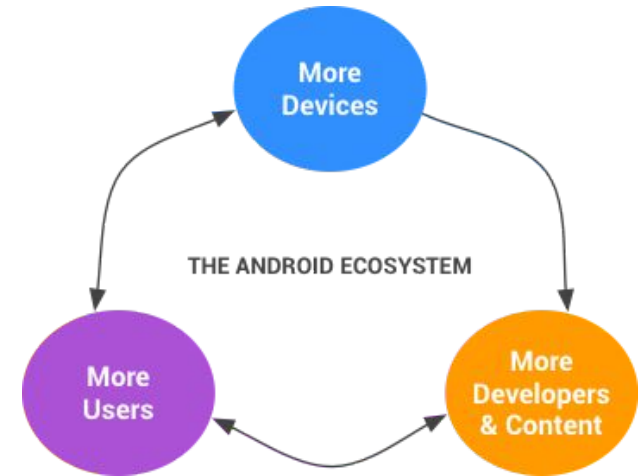
- › Binder IPC
- › art Java runtime
- › zygote
- › Wake Locks
- › bionic C library
- › HAL vs /dev
- › [...]



Woher kommt der Play Store?

AOSP != “powered by Android”

- › Google Apps, wie Play Store, Gmail, Chrome, Maps, Youtube
- › Google Play Services bzw. Google Mobile Services – GMS
- › Compatibility Test Suite – CTS
- › Android Compatibility Definition Document – CDD



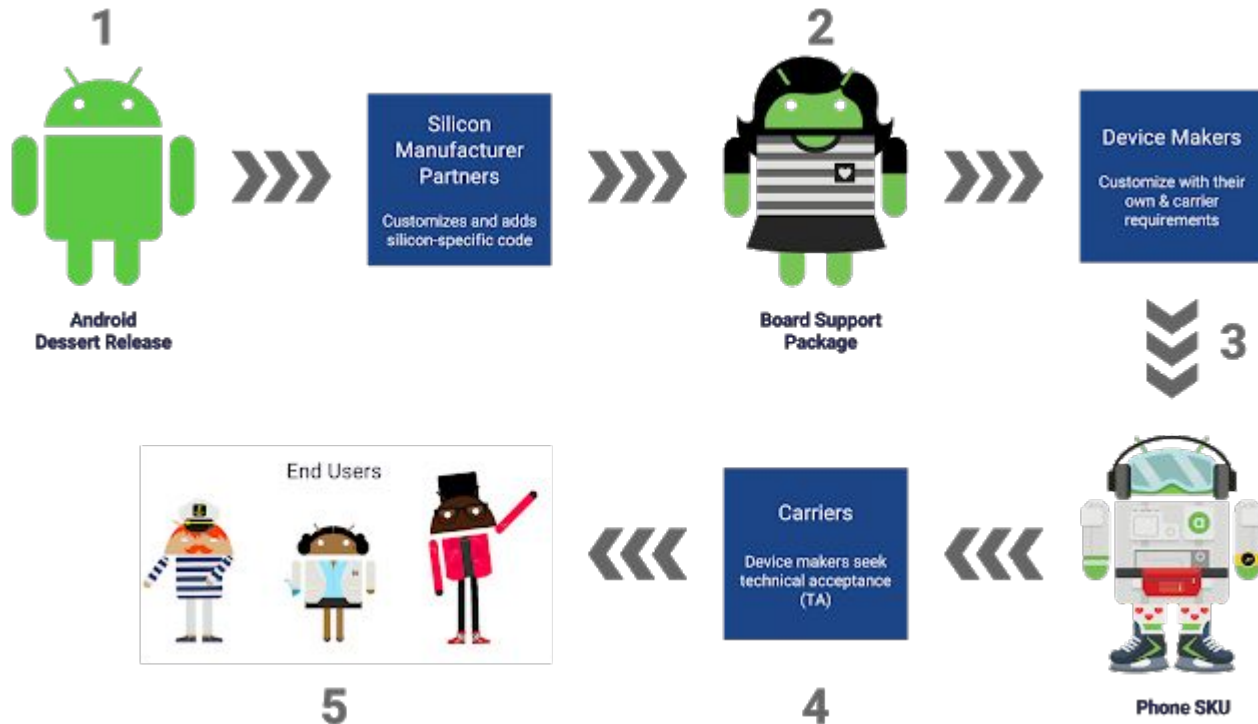
Das AOSP ist OpenSource,
aber Google kontrolliert
das (App-)Ökosystem.

Security Features

- › Kein Root Zugriff
- › Unix Dateisystem Rechte und Benutzer, SELinux für App-App Isolierung
- › App Signierung
- › App Permissions, Runtime Permissions (Android 6.0)
- › “Secure Boot” aka “Verified Boot”
- › Benutzerdaten-Verschlüsselung
- › Trust Zone, Trusty OS

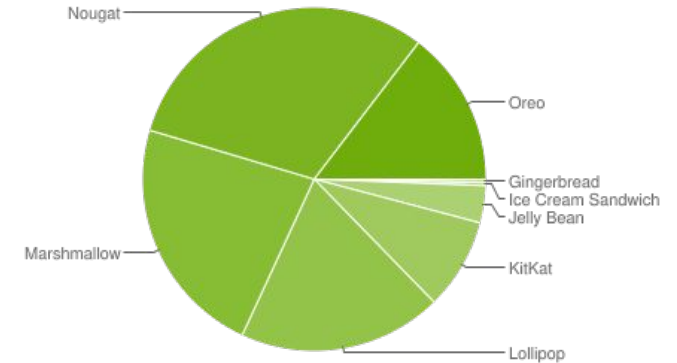
“Project Treble”

Wie kommt das AOSP auf das Smartphone



“Project Treble”

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	0.3%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0.3%
4.1.x	Jelly Bean	16	1.1%
4.2.x		17	1.6%
4.3		18	0.5%
4.4	KitKat	19	7.8%
5.0	Lollipop	21	3.6%
5.1		22	14.7%
6.0	Marshmallow	23	21.6%
7.0	Nougat	24	19.0%
7.1		25	10.3%
8.0	Oreo	26	13.4%
8.1		27	5.8%

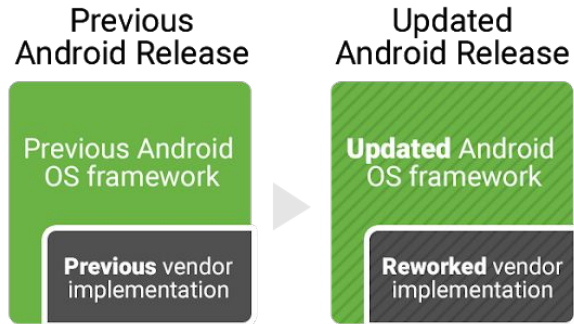


“Project Treble”

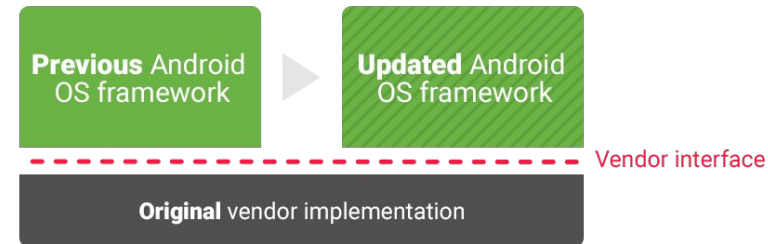
Updates direkt von Google

- › **HIDL** - “HAL interface definition language”

Before Treble



With Treble



2. Teil – Praxis

Books

for embedded Android development

- › Learning Embedded Android N Programming
Ivan Morgillo, Stefano Viola
Packt Publishing

<https://www.packtpub.com/application-development/learning-embedded-android-programming>

- › Embedded Android
Karim Yaghmour
O'Reilly

<https://www.oreilly.com/library/view/embedded-android/9781449327958/>

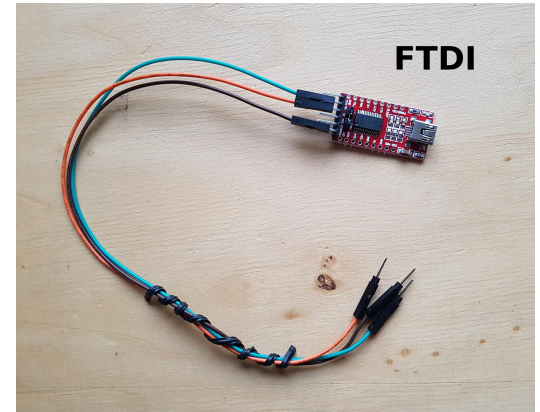
Hardware requirements

Workstation:

Quad Core - i7, 16 GB RAM,
SSD, 150 GB disk storage

Tools:

USB cables, FTDI for serial port access



Software Requirements

OS: Ubuntu or Mac OS

Tools:

- › fastboot
- › adb – Android Debug bridge

```
$ fastboot devices  
FA7C71A04090 fastboot
```

```
$ adb logcat *:S SurfaceFlinger:*  
07-09 06:01:31.718 612 612 I SurfaceFlinger: SurfaceFlinger is starting.  
07-09 06:01:31.722 612 612 I SurfaceFlinger: Enabling HWC virtual displays
```

Select a device

Best natively supported in the AOSP source tree

- › HiKey 960 – 96 Boards
- › Pixel 2 – Google



The local setup

- › Unlocked Bootloader



Demo - Goal

What we will do

1. Downloading the AOSP source tree
2. Downloading proprietary binary drivers
3. Make a very tiny modification to the source code
4. Building the AOSP and generation system partitions
5. Flashing the partitions onto your device
6. Boot it

Demo – Let's go

Cloning, building and flashing the AOSP

Start here:

<https://source.android.com/>

Demo – Commands

```
# apt-get install some stuff
$ mkdir ~/bin
$ PATH=~/.bin:$PATH
$ curl https://storage.googleapis.com/git-repo-downloads/repo > ~/bin/repo
$ chmod a+x ~/bin/repo
$ mkdir ~/tmp/aosp/
$ cd ~/tmp/aosp
$ repo init -u https://android.googlesource.com/platform/manifest -b android-8.1.0_r1
$ repo sync
# look at https://developers.google.com/android/drivers
$ https://dl.google.com/dl/android/aosp/google_devices-walleye-opm4.171019.021.q1-7d8aad26.tgz
$ tar xf google_devices-walleye-opm4.171019.021.q1-7d8aad26.tgz
$ ./extract-google_devices-walleye.sh
$ rm google_devices-walleye-opm4.171019.021.q1-7d8aad26.tgz google_devices-walleye-opm4.171019.021.q1-7d8aad26.tgz
$ wget https://dl.google.com/dl/android/aosp/qcom-walleye-opm4.171019.021.q1-f06ed7da.tgz
$ tar xf qcom-walleye-opm4.171019.021.q1-f06ed7da.tgz
$ ./extract-qcom-walleye.sh
$ rm extract-qcom-walleye.sh qcom-walleye-opm4.171019.021.q1-f06ed7da.tgz
$ source build/envsetup.sh
$ lunch
$ lunch aosp_walleye-eng
$ m -j8
$ adb reboot bootloader
$ fastboot flashall -w
```

Demo – Done!

Congratulations. Your first custom ROM.

```
$ adb logcat *:S SurfaceFlinger:*  
07-09 06:01:31.718 612 612 I SurfaceFlinger: SurfaceFlinger is starting. (Hello there!)  
07-09 06:01:31.722 612 612 I SurfaceFlinger: Enabling HWC virtual displays
```

Vielen Dank

Stefan Lengfeld
Android Embedded

inovex GmbH
Gebäude 1.13
Schanzenstraße 6-20
51063 Köln

stefan.lengfeld@inovex.de



Das Android Open Source Project

Oder “Alles, was du schon immer über die Software auf deinem Android Smartphone wissen wolltest.”

Das Android Open Source Project, kurz AOSP, ist das Betriebssystem, das auf den meisten heutigen und wahrscheinlich auch auf deinem Smartphone läuft. Es ist die Basis für das Android App Universum und wird von Millionen Nutzern und Entwicklern auf der Welt verwendet. Wegen der offenen Verfügbarkeit des Source Codes ist es auch die Basis für bekannte Custom ROMs wie LineageOS.

Der erste Teil des Talks gibt eine Übersicht über die Architektur des Betriebssystems, das App Ökosystem, den Hardware Abstraction Layer (HAL), die Sicherheitskonzepte und einige neue Betriebssystem-Entwicklungen, wie Project Treble in Android 8.0.

Im zweiten Teil des Talks schauen wir uns den Quellcode und die Struktur des AOSP näher an: Wie lädt man sich den Source Code herunter, wie baut man das AOSP für unterstützte Geräte und wie kann man die eigenen ROMs auf ein Smartphone flashen. Zum Spaß blicken wir vielleicht in einige Implementierungsdetails von Android App API Funktionen, die man als App Developer schon aufgerufen hat.