

# Is Android the new Embedded Linux?

**AnDevCon V**

**Karim Yaghmour**

karim.yaghmour@opersys.com





These slides are made available to you under a Creative Commons Share-Alike 3.0 license. The full terms of this license are here:  
<https://creativecommons.org/licenses/by-sa/3.0/>

Attribution requirements and misc., PLEASE READ:

- This slide must remain as-is in this specific location (slide #2), everything else you are free to change; including the logo :-)
- Use of figures in other documents must feature the below “Originals at” URL immediately under that figure and the below copyright notice where appropriate.
- You are free to fill in the “Delivered and/or customized by” space on the right as you see fit.
- You are FORBIDDEN from using the default “About” slide as-is or any of its contents.
- You are FORBIDDEN from using any content provided by 3<sup>rd</sup> parties without the EXPLICIT consent from those parties.

(C) Copyright 2013, Opersys inc.

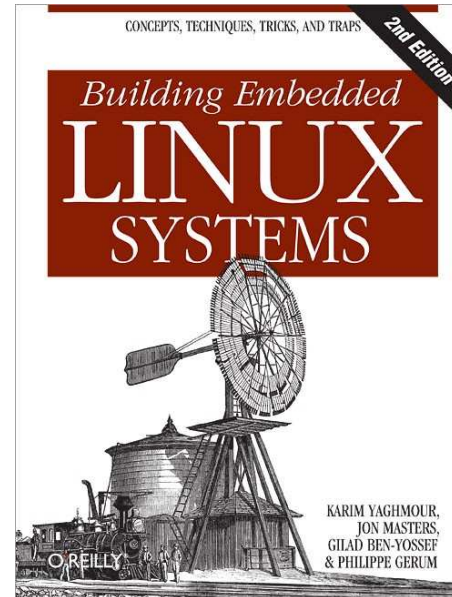
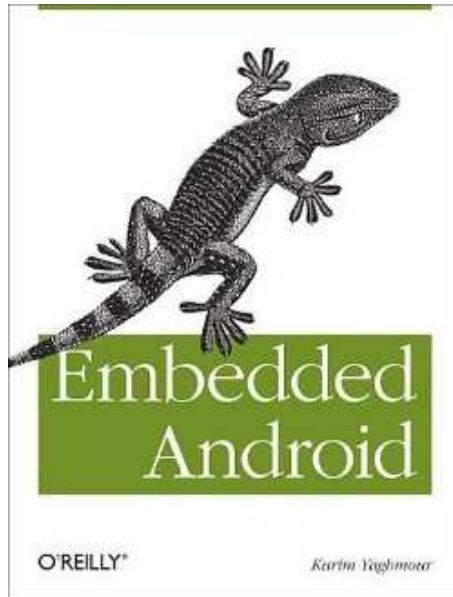
These slides created by: Karim Yaghmour

Originals at: [www.opersys.com/community/docs](http://www.opersys.com/community/docs)

Delivered and/or customized by

# About

- Author of:



- Introduced Linux Trace Toolkit in 1999
- Originated Adeos and relayfs (kernel/relay.c)
- Training, Custom Dev, Consulting, ...

# 1. Why are we asking this question?

- Android is based on Linux
- Android is “embedded”
- Android is extremely popular
- Android enjoys good support from SoC vendors

**Mostly - The trends are there**

# 1.1. Why did Embedded Linux rise?

- EETimes 2005 survey ...

<http://www.embedded.com/electronics-blogs/-include/4025539/Embedded-systems-survey-Operating-systems-up-for-grabs>

- EETimes 2013 survey

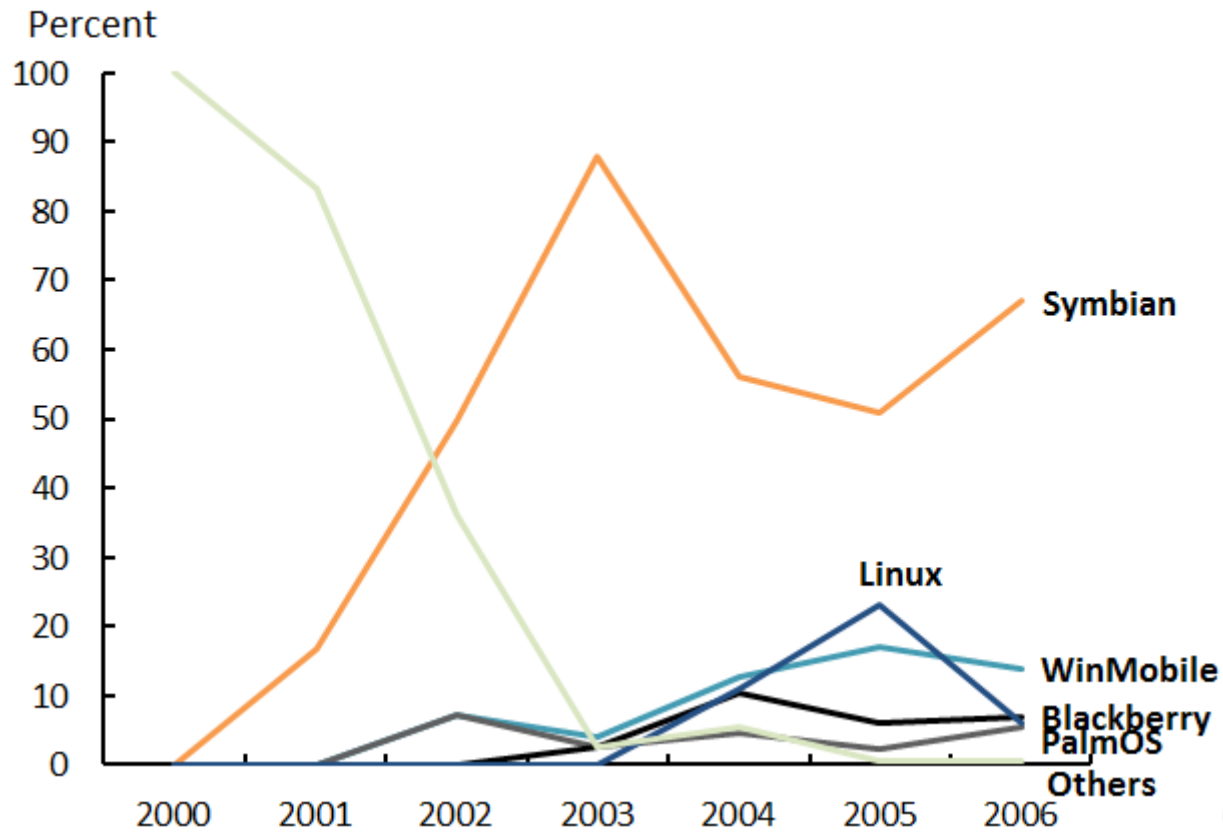
<http://www.slideshare.net/MTKDMI/2013-embedded-market-study-final>

<http://www.eetimes.com/electronics-news/4407897/Android--FreeRTOS-top-EE-Times--2013-embedded-survey>

# 1.2. Current consumer trends

- Rise of touch-based devices
- Race with Apple on features / price
- Mobile devices outselling PCs since 2010
- Increasingly inexpensive low-end touch-based devices

## Smartphone market share: 2000-2006



Source:

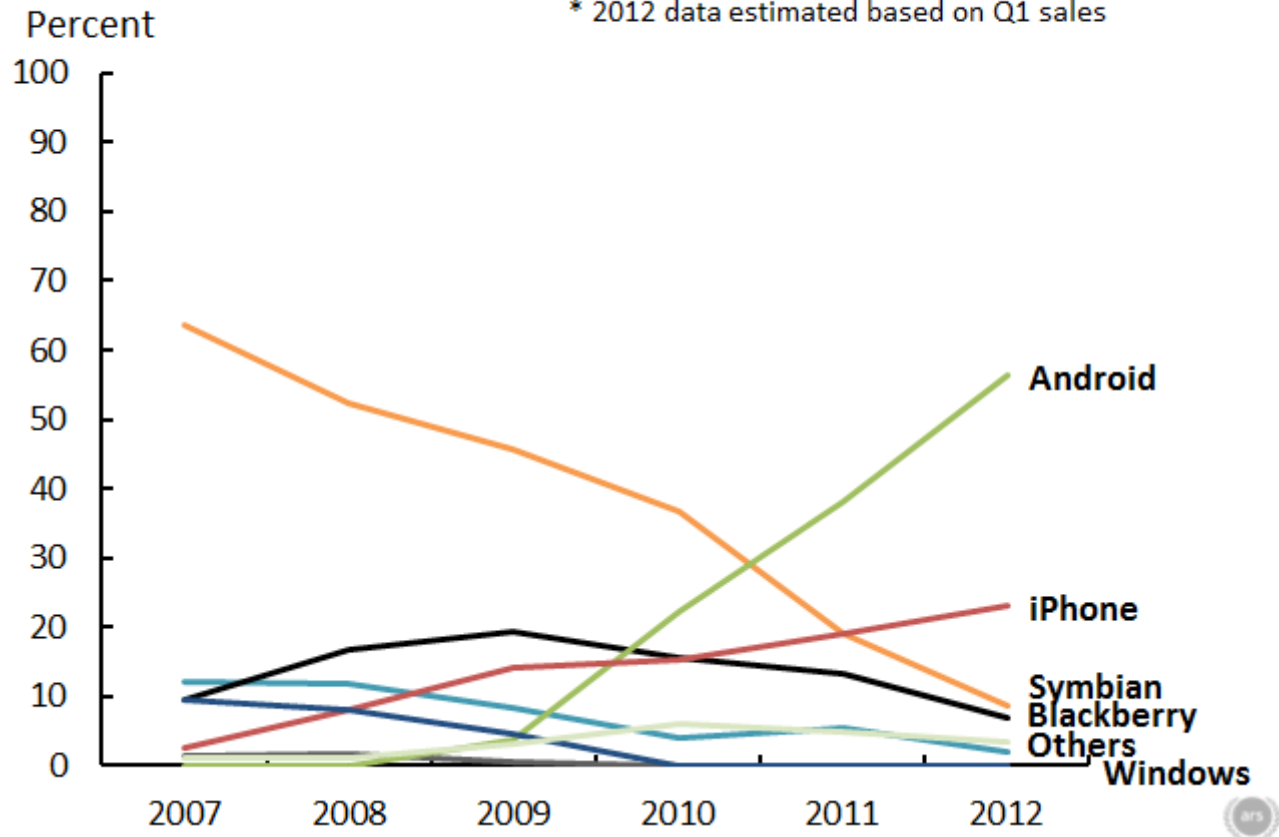
“From Altair to iPad: 35 years of personal computer market share”, Jeremy Reimer, Ars Technica, August 2012

<http://arstechnica.com/business/2012/08/from-altair-to-ipad-35-years-of-personal-computer-market-share/>

**This material is (C) 2012, Ars Technica and is NOT CC-BY-SA.**

## Smartphone market share: 2007-2012 \*

\* 2012 data estimated based on Q1 sales



Source:

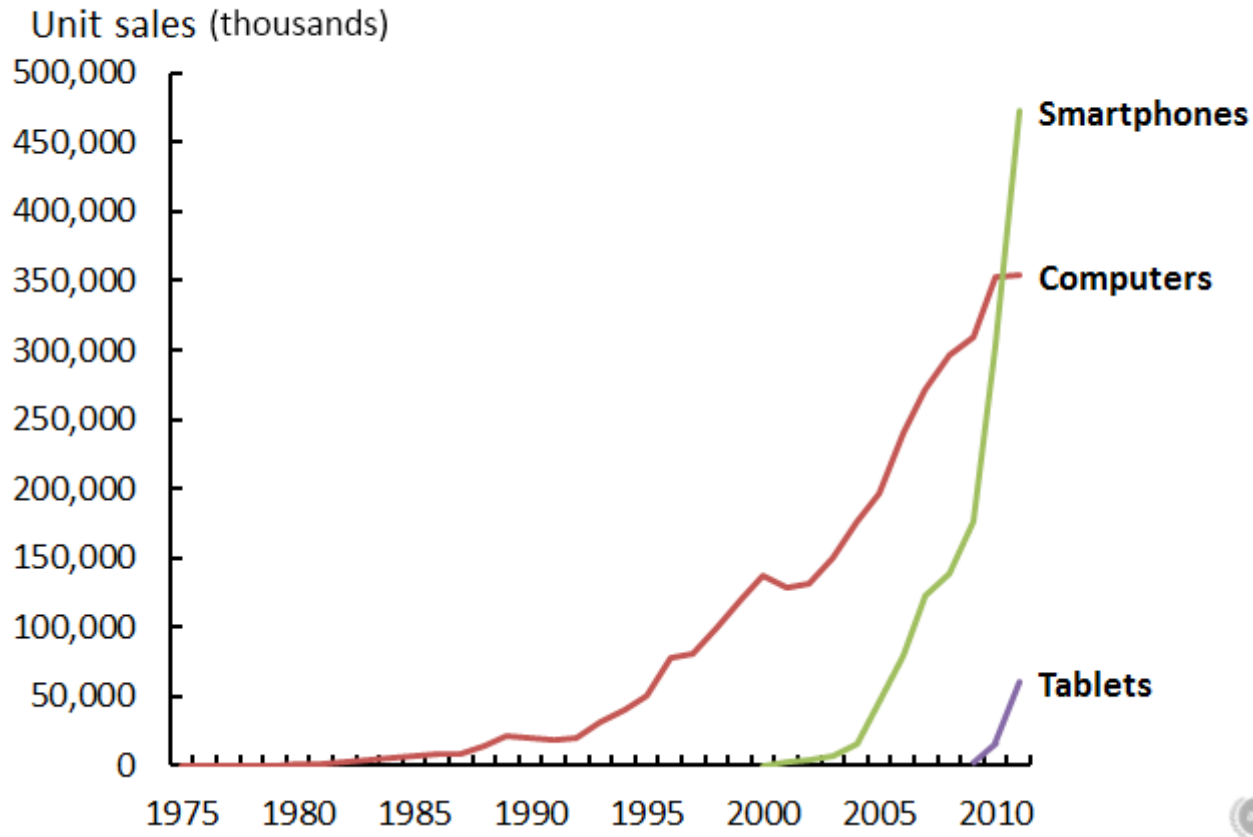
“From Altair to iPad: 35 years of personal computer market share”, Jeremy Reimer, Ars Technica, August 2012

<http://arstechnica.com/business/2012/08/from-altair-to-ipad-35-years-of-personal-computer-market-share/>

**This material is (C) 2012, Ars Technica and is NOT CC-BY-SA.**



## Computers, smartphones, and tablet sales: 1975-2011



Source:

“From Altair to iPad: 35 years of personal computer market share”, Jeremy Reimer, Ars Technica, August 2012

<http://arstechnica.com/business/2012/08/from-altair-to-ipad-35-years-of-personal-computer-market-share/>

**This material is (C) 2012, Ars Technica and is NOT CC-BY-SA.**

# 1.3. Current embedded trends

- Rise of cheap 32-bit CPUs
- Rise of inexpensive storage
- Rise of inexpensive eval boards
  - BeagleBoards
  - BeagleBone (Black)
  - RaspberryPi
  - For all practical purposes, this hardware is disposable
- Arduino / Maker / DIY trend
- UBM 2013 Embedded Market Study

# 2. Embedded Linux, a look back

## Projects

Linux kernel	1991 ...
RTLinux	1996 V. Yodaiken / M. Barabanov
uClinux / uCDimm	1998 J. Dionne / Motorola Dragonball
BusyBox	1999 B. Perens ('96) / Debian, then E. Anderson
RTAI	1999 P. Mantegazza
Buildroot	1999 E. Anderson
U-Boot	1999 W. Denk
uClibc	2000 E. Anderson
I-pipe / Adeos	2001 P. Gerum (K. Yaghmour)
Preemption patches	2005 S.-Thorsten Dietrich / I. Molnar / T. Gleixner
eglibc	2009 Debian
Yocto	2010 LF / Intel

## Attempts at Linux mobile GUIs

Maemo / MeeGo  
Matchbox  
Qttopia

## Conferences:

OLS	1999
CELF ELC	2004
ABS	2011

## Books

Craig Hollabaugh  
Karim Yaghmour  
Chris Hallinan

## Companies:

Lineo  
MontaVista  
TimeSys

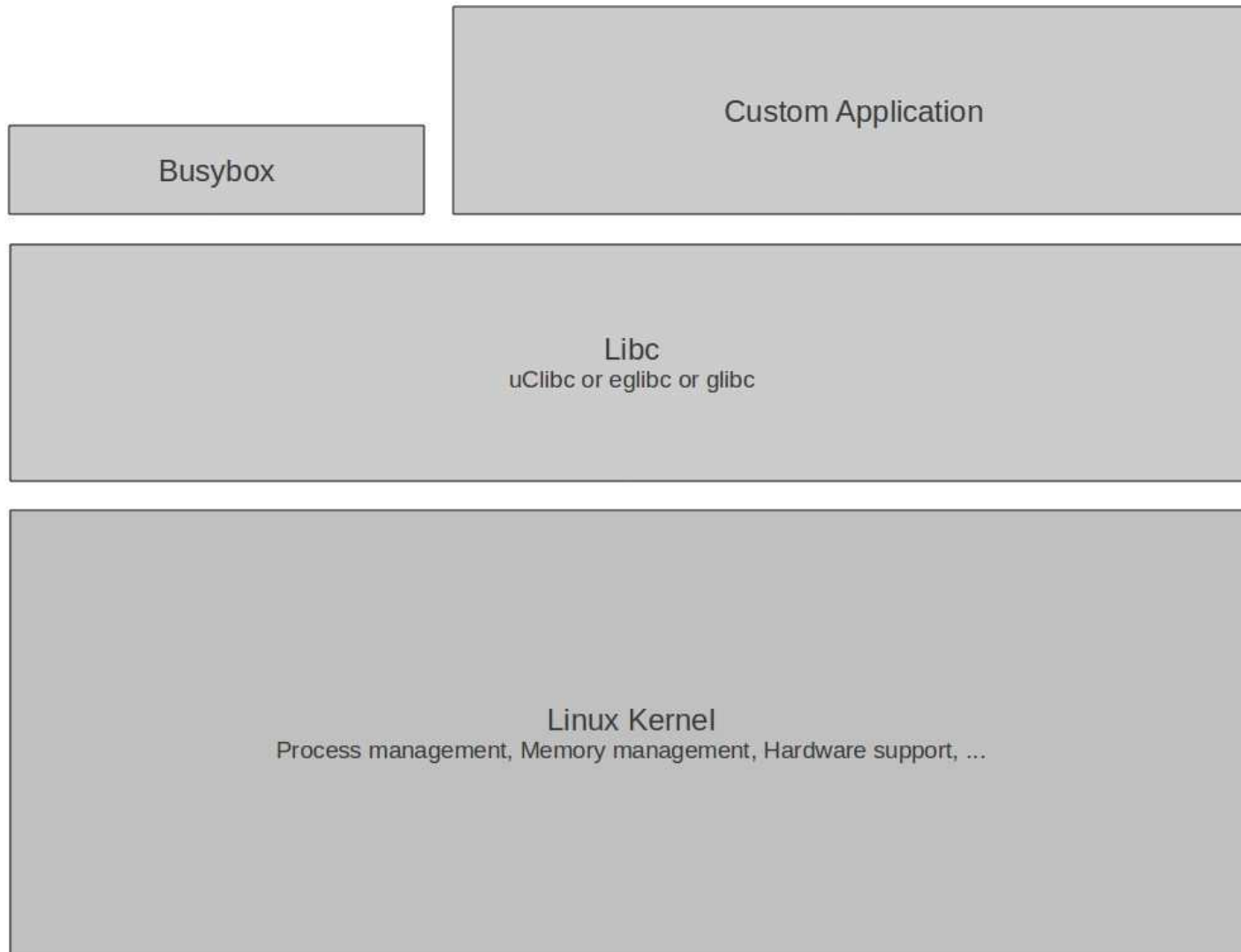
# 3. Android, whereth comest thou?

- 2002:
  - Sergey Brin and Larry Page started using Sidekick smartphone
  - Sidekick one of 1st smartphones integrating web, IM, mail, etc.
  - Sidekick was made by Danger inc., co-founded by Andy Rubin (CEO)
  - Brin/Page met Rubin at Stanford talk he gave on Sidekick's development
  - Google was default search engine on Sidekick
- 2004:
  - Despite cult following, Sidekick wasn't making \$
  - Danger inc. board decided to replace Rubin
  - Rubin left. Got seed \$. Started Android inc. Started looking for VCs.
  - Goal: Open mobile hand-set platform
- 2005 - July:
  - Got bought by Google for undisclosed sum :)
- 2007 - November:
  - Open Handset Alliance announced along with Android

# 4. What's Embedded Linux?

# 4. What's Embedded Linux?

- A set of ad-hoc methods to package the Linux kernel with a (minimal) filesystem.
- FS content “to be determined” case-by-case
- APIs are specific to each device/build
- “Core software”:
  - BusyBox
  - U-Boot
  - GNU Toolchain
- Your flavor of:
  - glibc or uClibc or eglibc
  - yocto or buildroot or eldk or ltib or ptxdist or ...
- No serious UX framework



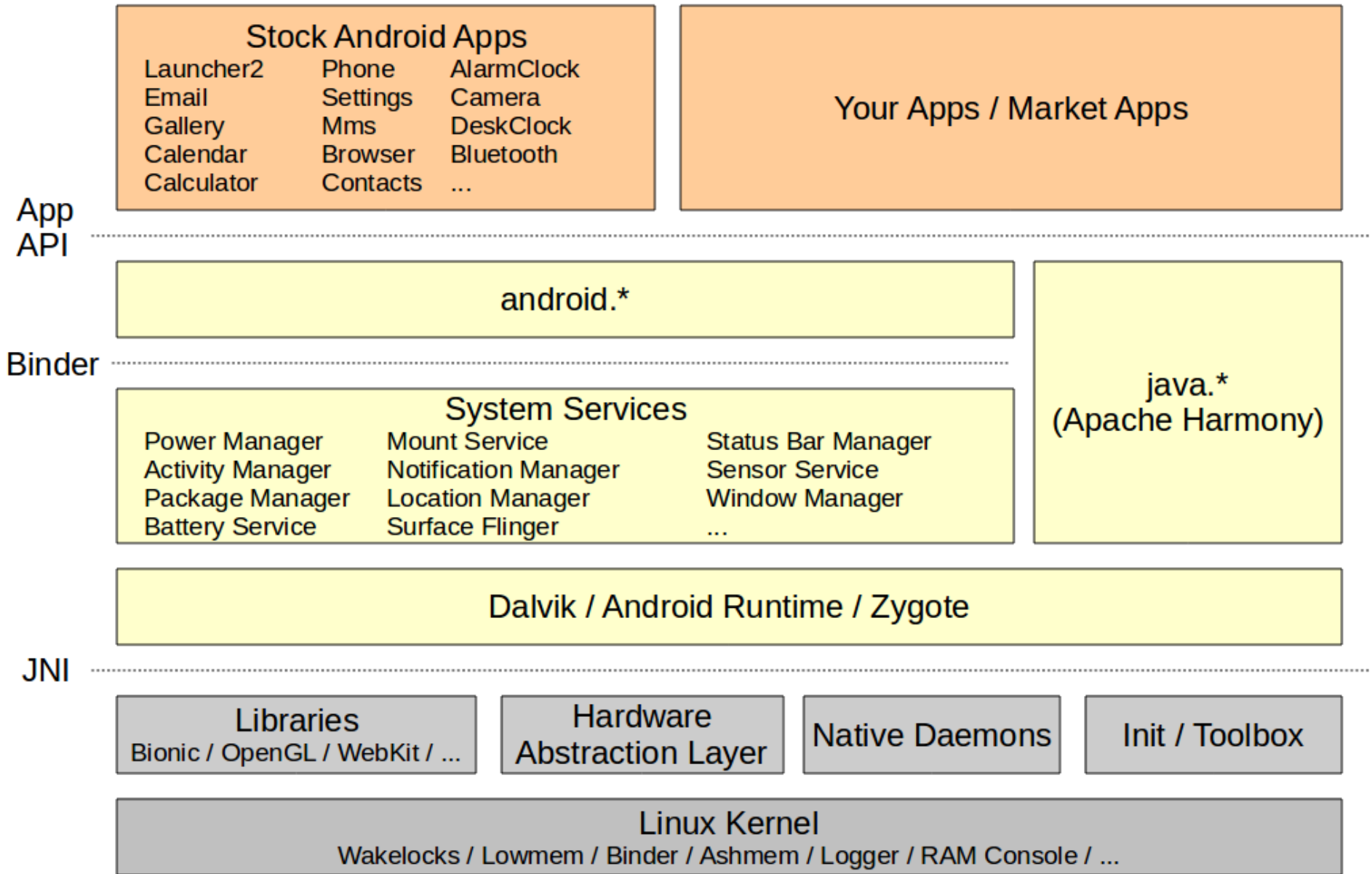
# 5. What's Android?



# 5. What's Android?

- ... apart from its increasingly well known UX ...
- Fully-integrated Eclipse IDE
- SDK/NDK
- ADB
- Fastboot
- Published, well-known, and very rich APIs
- A large and growing developer community
- And still we can use the usual suspects:
  - GNU toolchain, BusyBox, u- boot, glibc, ...

**An actual standardized dev. env. across all product lines**



# 6. Licensing differences

- Kernel:
  - GNU General Public License (a.k.a. GPL)
- AOSP:
  - Mostly Apache License 2.0 (a.k.a. ASL)
  - Having GPL-free user-space was a design goal
  - A few GPL and LGPL parts: mainly BlueZ and DBUS
  - Some key components in BSD: Bionic and Toolbox
  - “external/” directory contains a mixed bag of licenses, incl. lots of GPL
- Android Robot:
  - Very much like the Linux penguin
- Android name:
  - As descriptor only: “for Android”
  - Use for labelling device requires talking to Google

# 7. Why did Android use Linux?

“The Linux vs BSD decision was made before Google acquired Android.

While starting with NetBSD's kernel was tempting, the Linux kernel wasn't that much larger (one of the concerns), had quite a lot of support from silicon vendors and OEMs, and a larger developer community (easier to find folks with Linux kernel expertise).

The bright line between the kernel and userspace drawn by the statement at the top of the Linux COPYING file was certainly helpful in making this decision. Explaining that the kernel is GPLv2 and userspace is Apache2/BSD/MIT is something easily understood by OEMs, and most OEMs don't have a problem with the general idea that the kernel is not where you differentiate your products.”

Brian Swetland, LWN, March 2011

# 8. Embedded Android? Why?

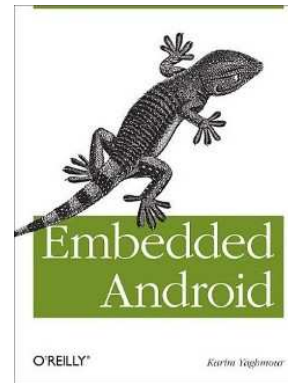
- Feature set
- User interface
- API
- Developer community
- App ecosystem
- Built on Linux
- Very strong SoC vendor support
- Permissive licensing
- Examples:
  - Consumer electronics
  - Medical
  - Defence
  - Space

# 9. Android's Challenges

- Documentation / Information
- Stack Limitations
- Stack Customization
- Boot time
- Legacy Integration
- Development Hardware
- AOSP tree
- Build Times
- Headless Use
- Dependency on Google

# 9.1. Documentation / Information

- Official doc for app dev is great
- Official doc for platform dev is practically non-existent
- A few official forums:
  - android-building
  - android-platform
  - android-porting
- Conference talks / presentations
- Scattered documentation all over the web
- My book (disclaimer applies):



# 9.2. Stack Limitations

- Key functionality not supported by default:
  - Ethernet
  - AM/FM radio
  - Real-time
- Supported functionality doesn't expose everything:
  - BlueZ
  - USB
  - wpa\_supplicant
- Bionic isn't a full glibc replacement:
  - No SystemV IPC
  - No locales
  - etc.
- Stack is very deep and can be difficult to modify



# 9.3. Stack Customization

- Lots of default packages
- Modularity wasn't *really* part of the design
  - No “make menuconfig”
- Can't easily select/change:
  - Which packages to include
  - Which system services are running
  - Packages' features
- Ex:
  - Can relatively easily replace default home screen
  - Modifying status bar requires digging in code

# 9.4. Boot time

- Very lengthy boot process
  - Bootloader
  - Kernel
  - Init
  - Zygote
  - System services
  - DEX optimization
  - Key apps
- Unlikely to be able to speed this through trimming ...
- Try:
  - Suspend to flash
  - Checkpointing

# 9.5. Legacy Integration

- What if I ... ?:
  - have been using my custom rootfs forever
  - really need glibc
  - have a huge stack based on “legacy Linux”
  - don't want to deal with AOSP's build system
  - want BusyBox
  - want to unify with a std Linux desktop
  - need things the AOSP doesn't provide
  - have a ton of Linux drivers and want those to be used by Android
  - ...
- Know this: filesystem integration is trivial:
  - Change build system to copy your custom rootfs
  - Change init scripts to start you own daemons
  - Use sockets to communicate
- Know this too: std Linux desktop integration is NOT

# Host

## AOSP (bionic)

build/  
system/  
frameworks/  
packages/  
external/

## PRJROOT (glibc)

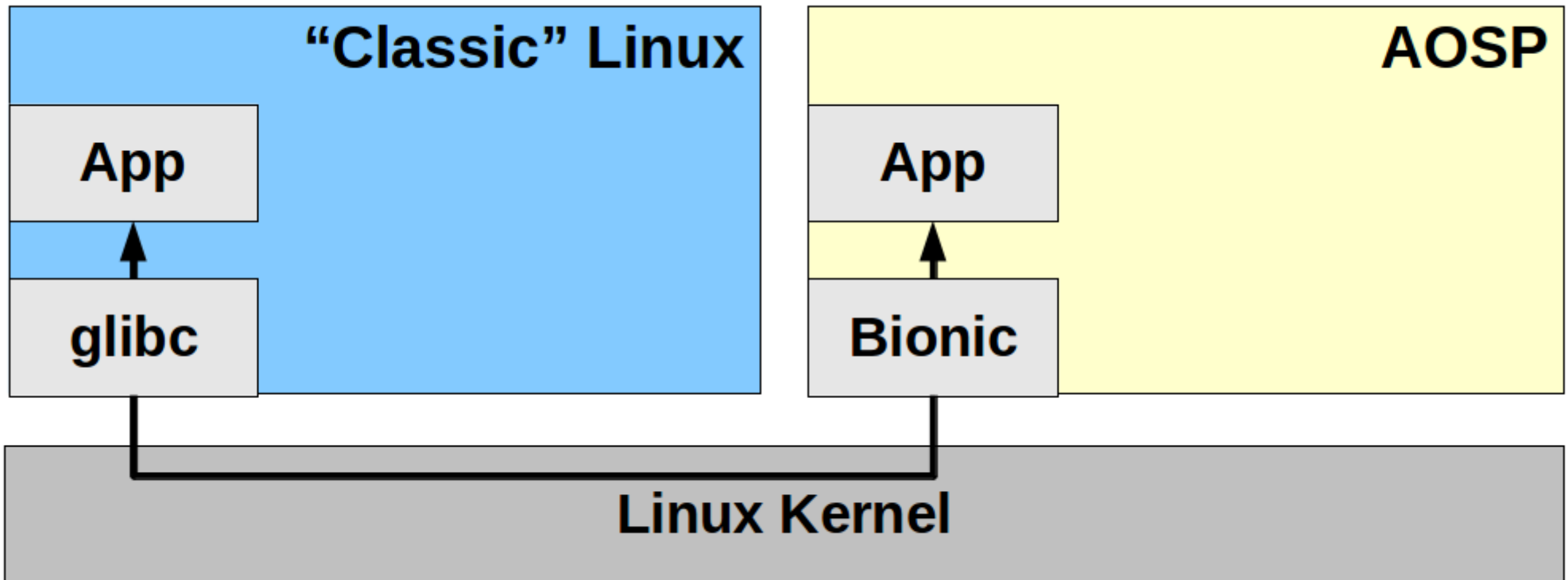
sysapps/  
rootfs/  
build-tools/  
.../

Kernel

# Target

- ramdisk.img => /
  - /bin
  - /lib
- system.img => /system
- userdata.img => /data

Kernel



# 9.6. Development Hardware

- Can I use a standard phone?
  - Not really
  - Even if you could – some things can only be done in the platform
- Which board is best to evaluate Android?
  - BeagleBone / BeagleBoard
  - PandaBoard
  - OrigenBoard
  - iMX
  - DragonBoard
  - AM335x Start Kit
- Still:
  - Nothing that has a phone/tablet form-factor + is entirely open

# 9.7. AOSP tree

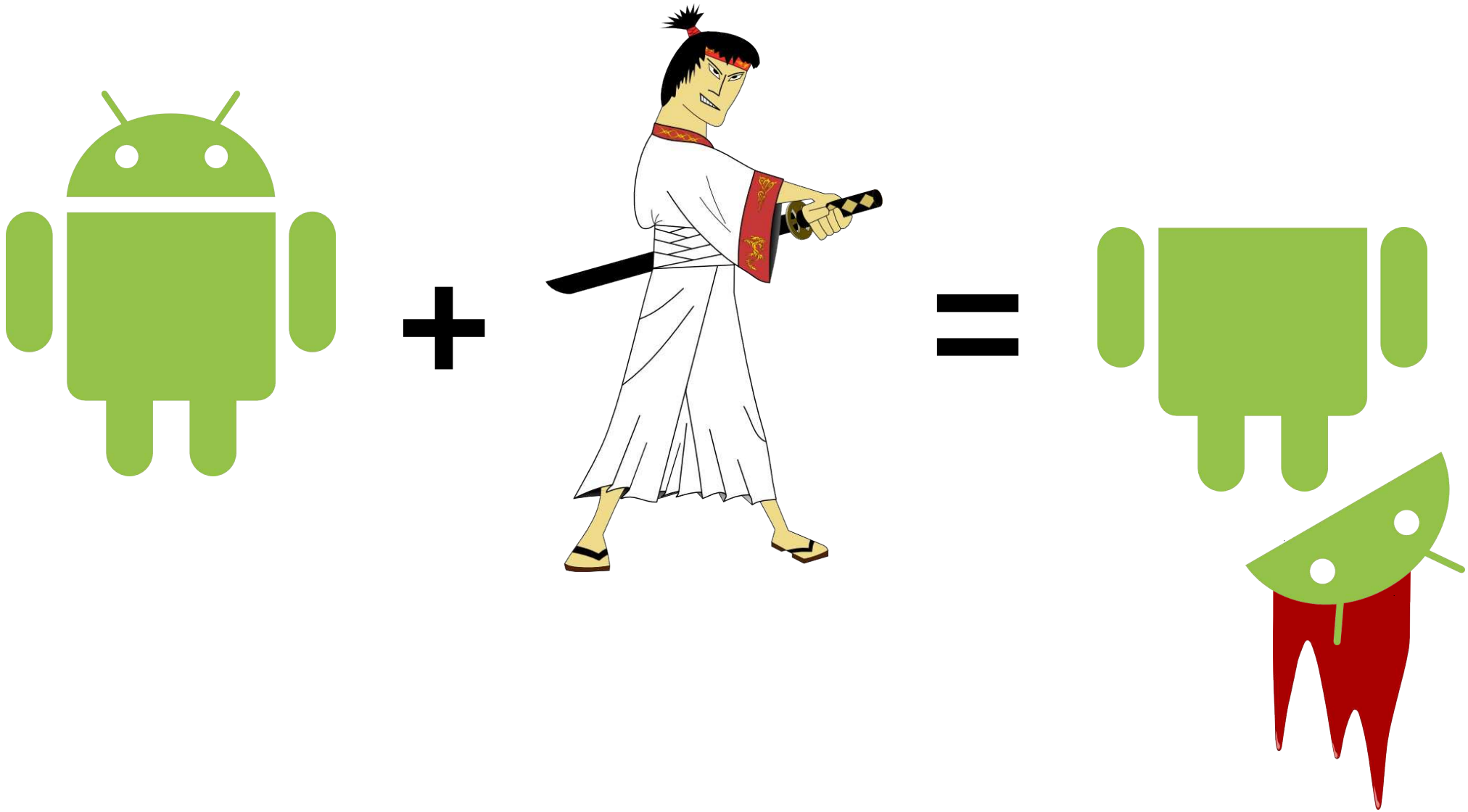
- Which tree?
  - Google's
  - Linaro's
  - TI's
  - Freescale's
  - Qualcomm's
  - etc.
- ... which kernel ... ?

## 9.8. Build Times

- Horrible
- On quad-core i7 w/ 8GB RAM:
  - GB = 20min
  - ICS = 55min
  - JB = 75min
- Even Google's own guys are at around 30min for JB.
- This is a big, unresolved pain



# 9.9. Headless Use



“And then GNU came back for revenge ...” -- Tarantino's sequel

## 9.9.1. Why?

- Took me a long time to wrap my *head* around
  - “Why don't you just use embedded Linux?”
- What's “Embedded Linux” anyway?
- NEW: Nexus Q

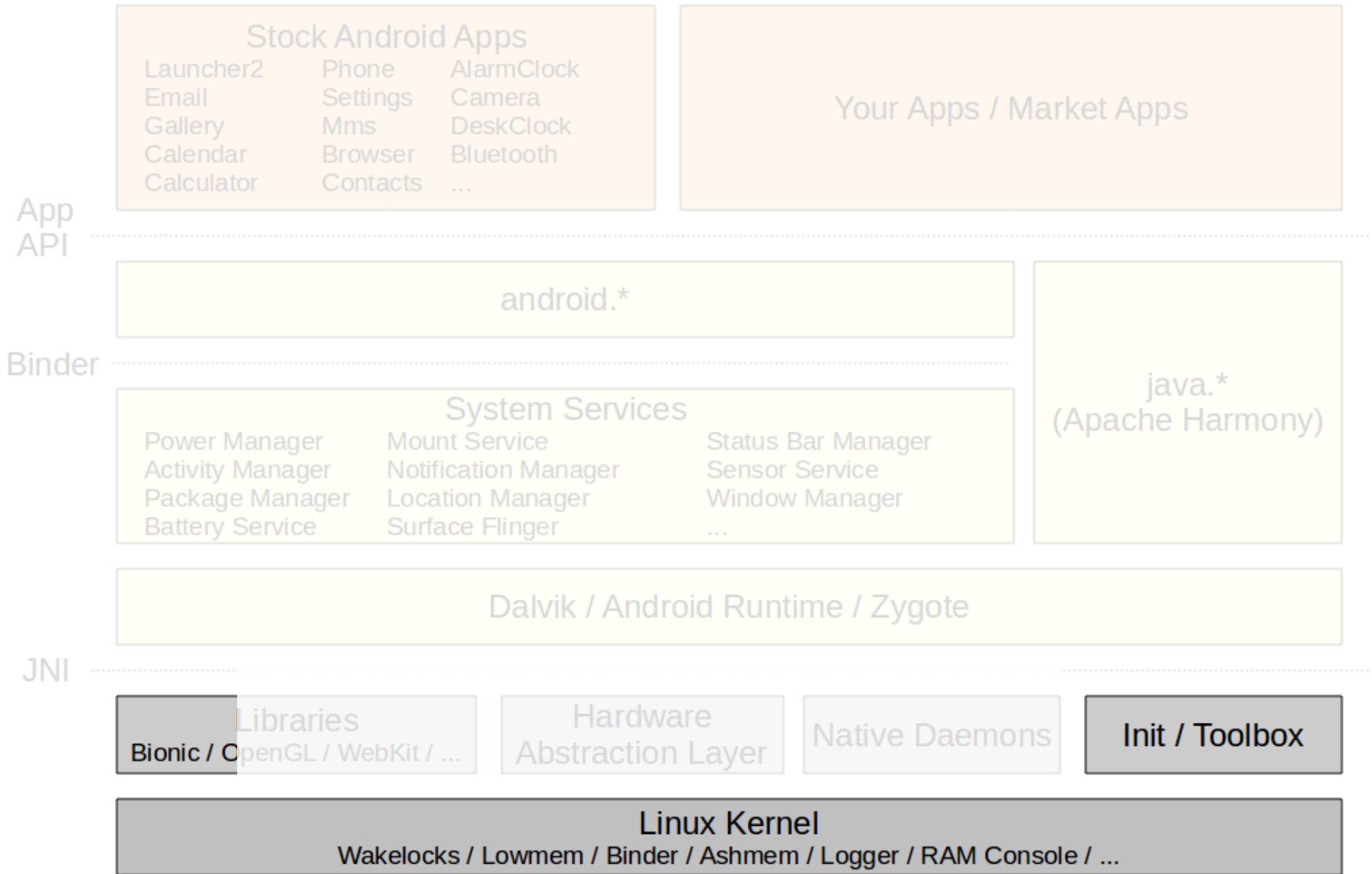
## 9.9.2. Possibilities

- No Java:
  - TinyAndroid:

```
$ BUILD_TINY_ANDROID=true make -j4
```
  - AOSP w/ custom products .mk file
- The full-blown stack **without**:
  - SurfaceFlinger
  - WindowManager
  - WallpaperService
  - InputMethodManager

## 9.9.3. Tiny Android

- 3MB filesystem
- Minimal root fs
- init
- toolbox + shell
- adb
- bionic + utility libs
- No “system/framework/”
- No “system/app”

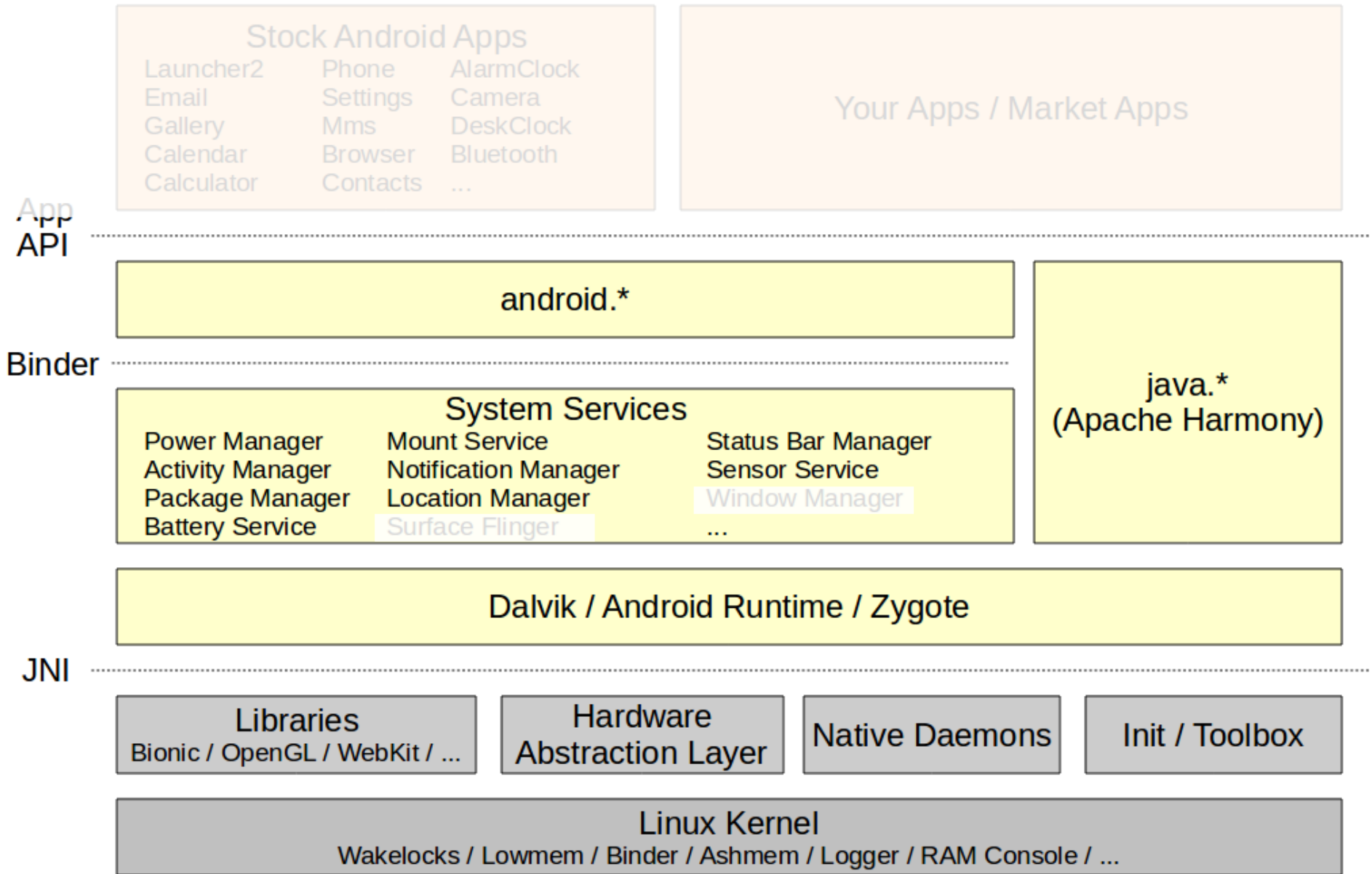


## 9.9.4. AOSP w/ custom product .mk

- Have a look at:
  - build/target/product/\*.mk
- Create your own device under “/device” and have fun
- Disable zygote at startup
- Remove all apks
- ...

# 9.9.5. Full stack

- Disable:
  - SurfaceFlinger
  - WindowManager
  - WallpaperService
  - InputMethodManager
  - SystemUI
- Don't let SurfaceFlinger Client try to open binder to SurfaceFlinger
- Feed bogus values back from SurfaceFlinger Client
- Disable qemud (emulator artefact)
- Tweak internals by disabling key calls:
  - In ActivityStack.java:
    - startHomeActivityLocked()
    - setAppStartingWindow()
  - wm.detectSafeMode()
  - wm.systemReady()
  - wm.reclaimSurfaceMemoryLocked()





- Demo
- Patch walkthrough
- `ro.config.headless` property

# 9.10. Dependency on Google

- Google is the master of Android
- Google's business prerogatives may not be yours
- Most new features come from Google
- Upstreaming is tough/lengthy
- What if Google stops developing Android?
  
- Still, and this is important: they like to see Android used in embedded systems.

# 10. Where do we go from here?

- Trends don't seem to be slowing down
- Interesting:
  - Nexus Q - headless from Google
    - i.e. Google moving to “true” embedded?
  - Ubuntu Tablet - SurfaceFlinger, not X nor Wayland
    - i.e. Std Linux moving to use Android parts?
- We're very much in uncharted waters

Thank you ...

[karim.yaghmour@opersys.com](mailto:karim.yaghmour@opersys.com)

