

Other Android community projects

Bernhard "Bero" Rosenkränzer, LCU 2013



Other Android community projects

- The semi-open nature of Android releases has caused numerous communities to create their own builds and forks
- There's currently not enough communication between the various communities – something we may want to try to fix
- For each community we're talking about, I'm going to list what we may want from them, not so much what they may want from us – because that's the same in every case: optimizations, support for newer toolchains, etc.
- Let's start with an overview of important projects:



Linaro Android

- Main focus: improvements for running Android on ARM
- Adds and improves support for various development boards (e.g. Panda, Origen, Arndale), Vexpress, FastModels – also runs on Nexus devices
- Adds ARM specific improvements, such as faster string handling routines in Bionic
- Adds generic improvements, such as support for newer toolchains, optimizations that aren't specific to ARM such as sincos() in Bionic
- Provides a development environment (gcc/make/... on Android)



01.org

- Main focus: improvements for running Android on x86
- Adds and improves support for x86 based development boards and reference devices
- Adds x86 specific optimizations
- Adds generic improvements such as Ethernet support, automatic kernel module loading (this will become relevant to us with better single zlmage support – definitely on ARMv8)
- Temporarily suspended because of merge with an internal Intel Android project



01.org – things we may want to copy

- Ethernet support (improved version of the Ethernet Connection Manager we already include)
- Automatic kernel module loading
 - Upstream Android wants to kill kernel modules
 - But that's a bad idea[™] as soon as we're talking about supporting any hardware that isn't guaranteed to remain constant (Aarch 64 boxes with PCI/PCI-E slots, development boards that have USB ports, ...)
- Optimizations that aren't specific to x86



android-x86.org

- Main focus: running Android on generic x86 PCs
- Adds and improves support for x86 devices
- Adds/improves support for PC type hardware such as mice, mouse wheel, external monitor/storage/keyboard/...
- Adds an installer





android-x86.org – things we may want to copy

- Improved mouse support etc. is relevant for all nontouchscreen devices
- Installer may become relevant if ARM devices manage to move into the generic computing space currently dominated by x86



cyanogenmod.org

- Main focus: Support as many phones and tablets as possible, generic improvements
- Currently supports 212 devices officially, more than 50 other devices supported unofficially
- Theming support, FLAC codec support, OpenVPN client, CPU overclocking support, interface modifications
- Assorted optimizations, including Linaro's string handling routines, support for Linaro toolchains (copied from Linaro Android)





cyanogenmod.org – things we may want to copy

- Device support would be nice to get more testing...
- Optimizations that affect ARM devices (and aren't copied from our tree in the first place)



Replicant

- Main focus: Being 100% Free/Open Source Software
- Removes all non-free components (even if it means losing functionality) and replaces them with free alternatives where possible



Replicant

Replicant – things we may want to copy

• Free replacements for components that provide all needed functionality – e.g. the Galaxy Nexus RIL layer may be interesting



Vendor branches

- Various chipset makers and handset makers make branches of Android optimized for their devices available, e.g.
 - **Sony**: http://developer.sonymobile.com/knowledge-base/open-source/android-open-source-project-for-xperia-devices/
 - Samsung: http://opensource.samsung.com/
 - **Allwinner**: https://github.com/allwinner-dev-team
 - Qualcomm: https://www.codeaurora.org/patches/quic/la/
 - Insignal (Arndale): http://git.insignal.co.kr/
 - Freescale:
 http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=IMXANDROID
 - ...



Vendor branches – things to copy

- Most vendor branches are a mess often with hundreds if not thousands of patches applied on top of an old release of AOSP
- But they often contain some very generic optimizations and bugfixes that should be upstreamed...
- Essentially, what Linaro does for the kernel would be needed here... But our Android team is nowhere near the size of the landing teams and most owners of vendor branches aren't our members



The future

- We should find ways for many if not all of the communities to work together and share generally useful work
- A combined effort may be better at attracting upstream attention



