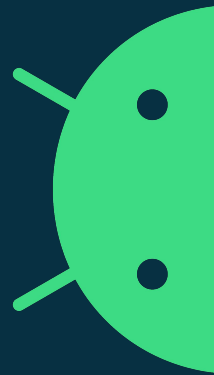


Android's Dream of a Stable Toolchain

LLVM Distributor Conf 2021

Pirama Arumuga Nainar pirama@google.com

Stephen Hines srhines@google.com



Who are we?

- Android toolchain maintainers
- Small team chasing ToT LLVM
- Bring LLVM goodies to Android
 - PGO/AutoFDO
 - clang-tidy, static analyzer
 - coverage
- Clang Built Linux
 - First and largest shipping user of this!

For whom?

- Users:
 - All native code in the Android platform since 2017
 - including Clang tools used during build
 - Kernel (Clang Built Linux)
 - Bootloader, CHRE
 - Native app developers (1p & 3p) (via Android Studio and NDK)
 - Tools like emulator, SDK, OpenJDK, etc.
- Platforms supported:
 - Linux, MacOS, Windows

What?

- Projects:
 - llvm;clang;lld;
 - libcxxabi;libcxx;compiler-rt;
 - clang-tools-extra;lldb
 - openmp;polly
- Runtimes required on device (cross-compiled)
 - compiler-rt, libunwind, openmp, lldb-server
 - targets: aarch64, arm32, x86_64, i386

Build process

- Not pure upstream ToT
 - need ~6 [downstream](#) changes
 - reverts
 - expose libclang c++ bindings
- Python [builder](#) to invoke multiple CMake commands
- Runtimes built using just-built stage2 toolchain
 - several variants targeting different Android API levels
- PGO + Thin LTO
 - PGO profiles generated and checked-in separately
 - PGO targets: Mix of ARM and X86, C/C++

Release process

- Work with ChromeOS toolchain team to pick revision
 - Share bug triage, cherry-picks etc.
 - Baseline typically validated by Google toolchain folks too
- [Revert checker](#) warns/alerts about reverted commits in upstream
- New release every 4-6 weeks as a team member rotation
 - Some releases have longer support window (NDK LTS)
-

Continuous Testing

- Android build uses `-Wall -Werror`
 - new/stricter warnings become errors
- Use a wrapper for clang, and clang-tidy to handle new warnings:
 - Run clang with original command
 - Report the errors and
 - ... re-run with `-Wno-error`
- Scripts to report new warnings by parsing logs