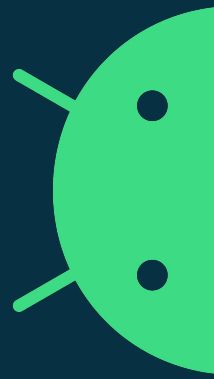
Android Engineering Productivity



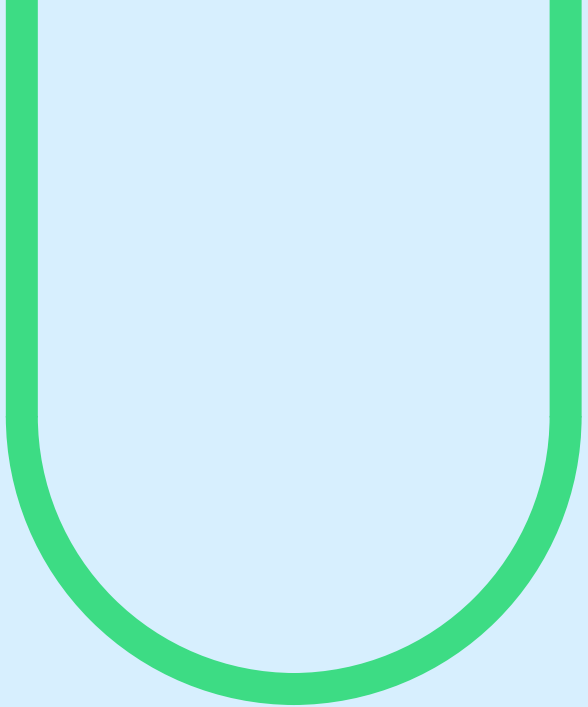
Tools & Infrastructure Offerings for Partners

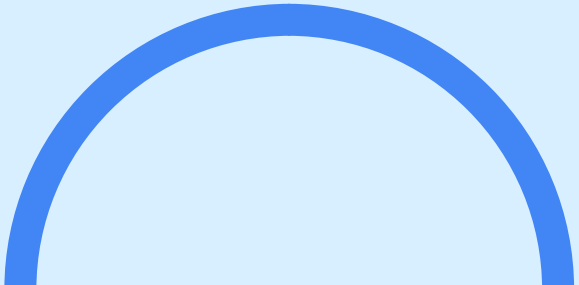
asillins@, larsun@, fdeng@, jeffbailey@

Agenda

- Background
- Tool offerings
- Partner interest
- Q&A and feedback

Background





android

Who We Are and what our expertises are

We are the Android Engprod team. We build development/test infrastructure and tools that allow Android to continuously deliver the best and the latest experience to users.

Expertise

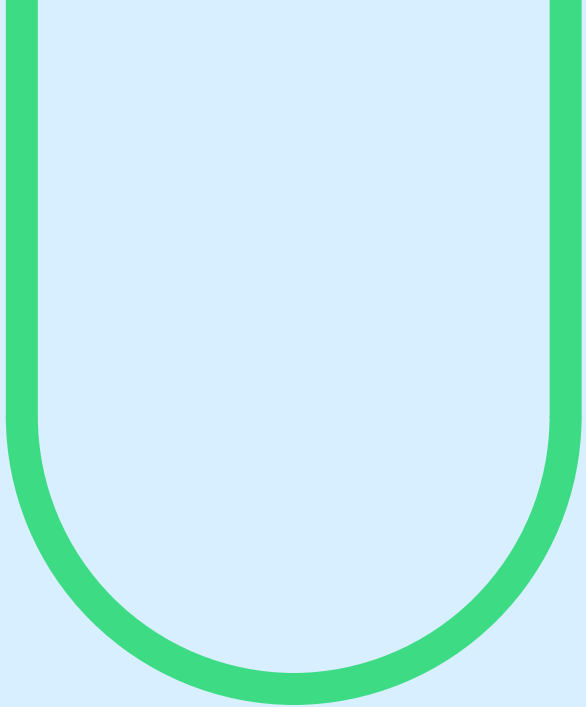
- *EngProd as a discipline at Google focuses on development velocity, quality, release, and monitoring*
- *The team builds tools to*
  - *Uncover quality issues upstream through manual + automated testing*
  - *Provide world-class tools to scale product engineers' feature iteration velocity (e.g. build, implementation, debug, etc.)*
  - *Shed light on issues through analytics and monitoring*

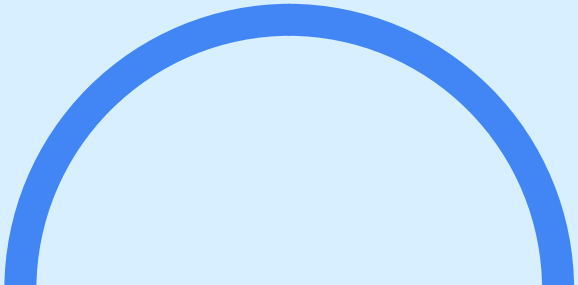
Our Goals

Our Goals:

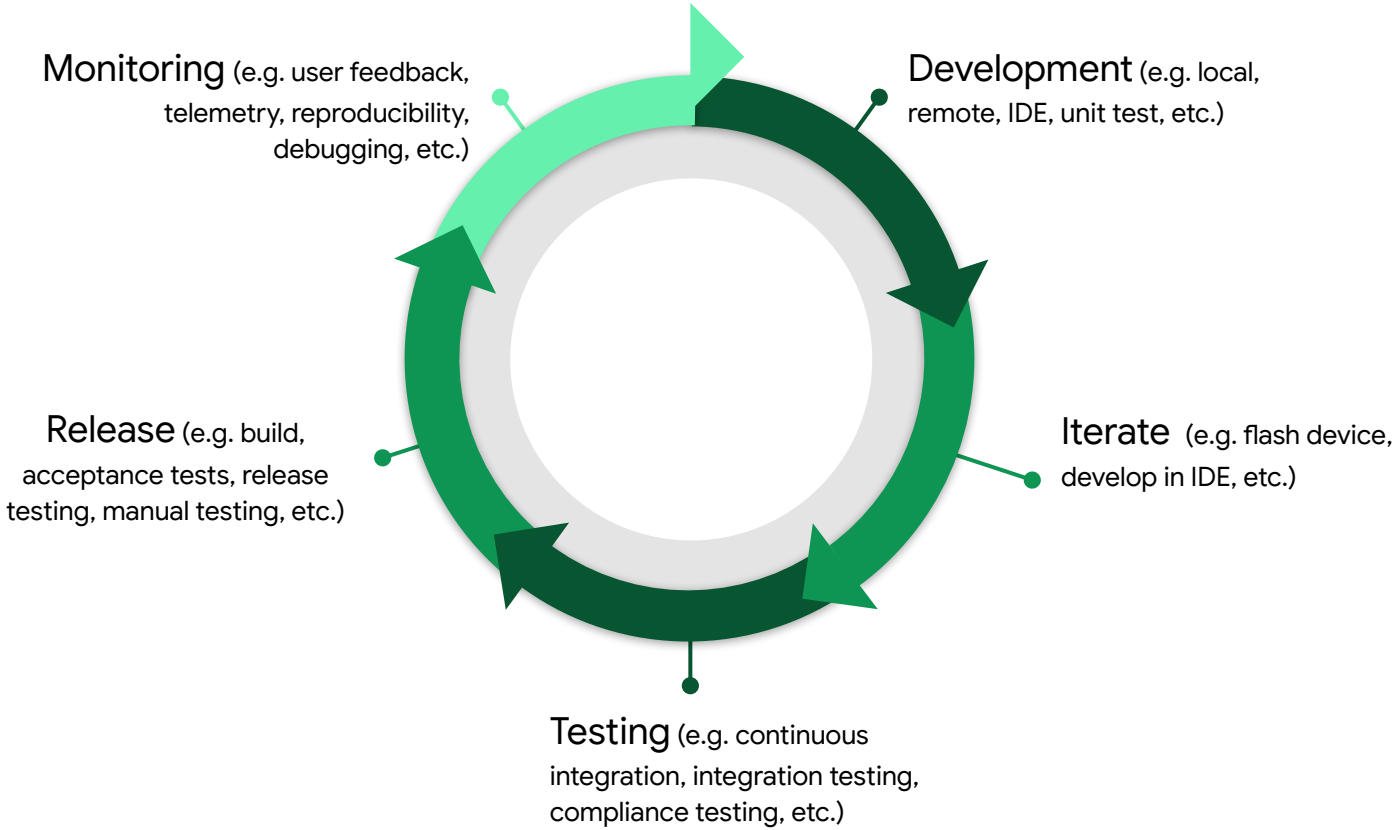
- From years of practicing EngProd, we think that we can share our practices to benefit the ecosystem and partners
- Understand the specific opportunity/challenge on assisting partners to **reduce cost**, and **improve velocity and quality** via externalization and/or collaboration on tools and infrastructure
- Equally interested in learning best practices from partners and your business/development workflow, to help Google to **improve quality at the source**

Tool Offerings

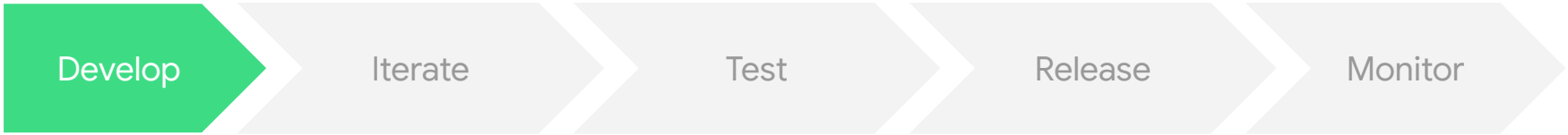




Development Workflow



android



Tool	Wins	Details
AIDEGen: set up project within the IDE	• Simplify the project setup • 1-stop for build, test, code review • More time to spend on “development”	• Generate IDE (e.g. Intellij, Android Studio) project for Android Platform developers • Enhancements that integrates continuous integration systems, Gerrit, various plug-ins
Soong: local build	• Make local build time reasonable • Enable iterative development	• Build system that allows engineers to build in a timely manner at local workstation
RBE: remote build execution	• xx% reduction in local (e.g. Developer local workstation) and remote (e.g. Continuous Integration) build time	• Remote build with caches across builds to reuse existing artifacts, significantly reduce build time

android

Develop | **Iterate** | Test | Release | Monitor

Tool	Wins	Details
Flashstation	• Easy integration with internal tooling • Flash latest Android AOSP builds	• Rated #1 internal Android tool QoQ
ACloud: developer and iterate against an AVD	• Reduce development cost with physical devices • Remote development and testing	
Iterating against a physical device	• N/A	• <Would like to get input for opportunity assessment/prioritization>
ATest: Local Testing	• Improve incentives for better quality at head • Enable test driven development	• xxx bugs prevented a month
Code search with cross references	• Navigate complex code base	• Code indexing / viewing / searching/ x-referencing / change reviewing • “Metadata”, lint errors, dead code analysis, code coverage, etc.

Public flashstation

Features

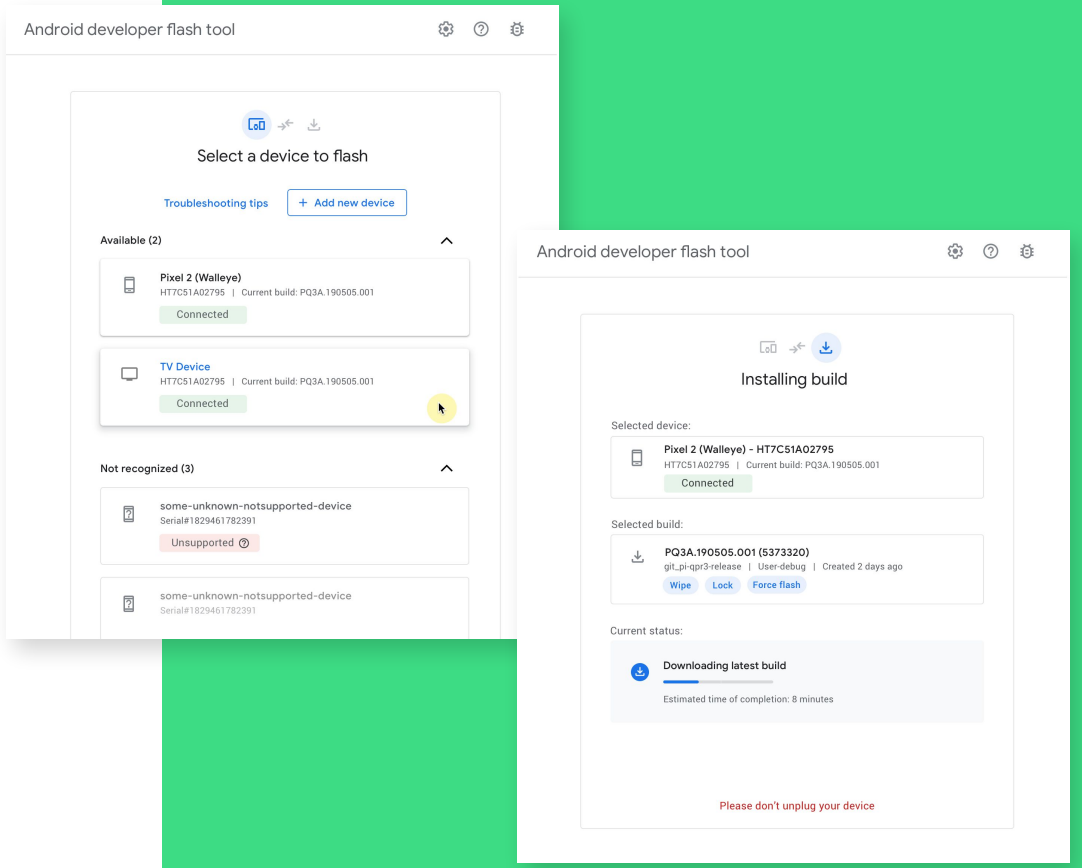
- Easily flash your Pixel or hikey device to any aosp Android version listed on ci.android.com

Use cases

- Android platform developers can test aosp builds
- App developers can test apps against the latest version of Android

Q2'2020

- Ability to flash a released build* onto a Pixel device
  - * those listed today on developers.google.com/android/images



android

Public flashstation

Open questions for partners

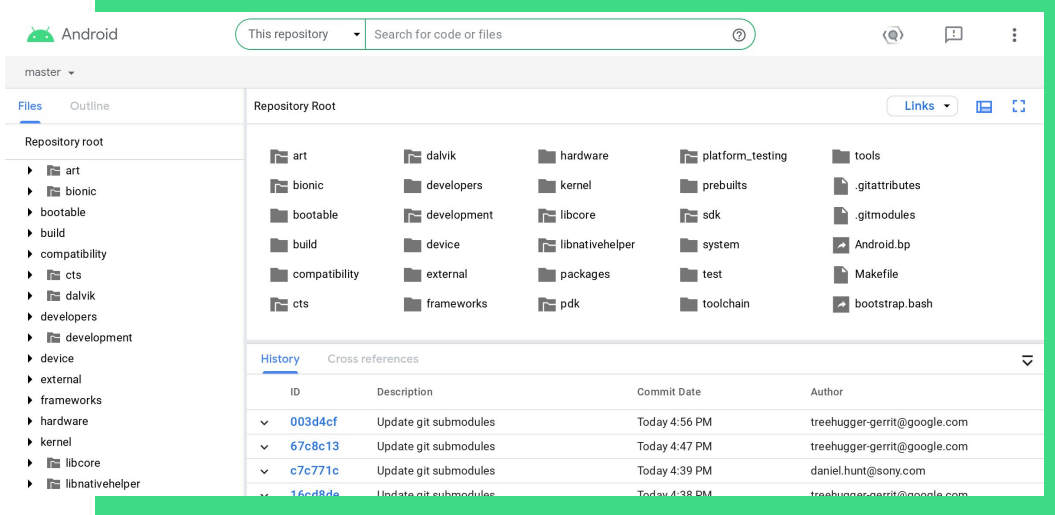
- From 1 to 10 (1 being the least, 10 being most), how interested are you in flashing builds from partner branches onto a pixel device?
- From 1 to 10 (1 being the least, 10 being most), how interested are you in support for flashing builds onto your own devices?
    - Gauging initial interest, as a change here might require device changes

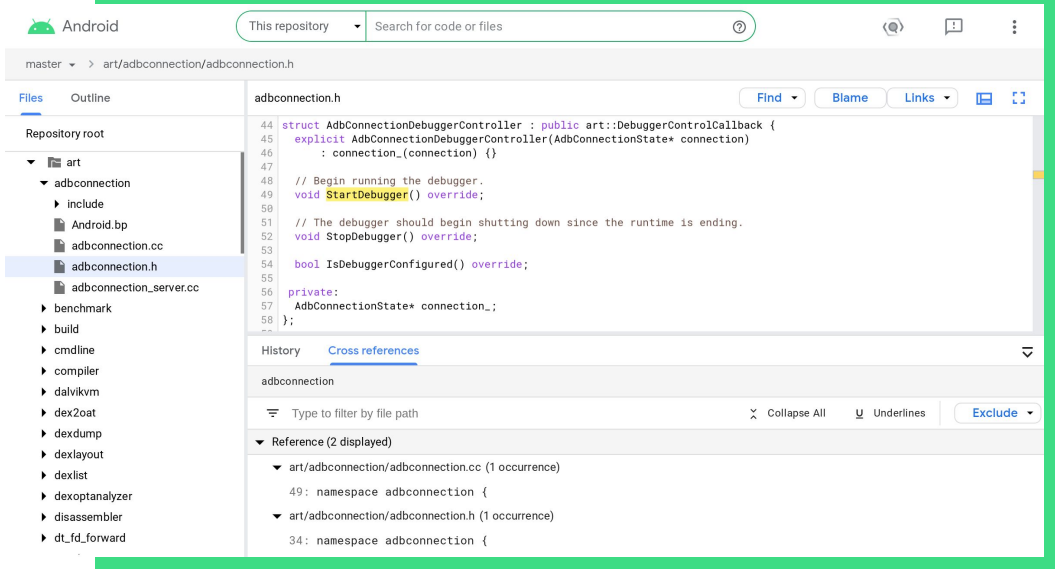
Code search with cross references for aosp

Features

- View and search the aosp source code locally without having to download the source
- Navigate cross-references across the entire code base that allow you to click through one part of the source code to another
- Switch between Android's open source branches

ETA: later this quarter





Code search with cross references

Open questions for partners

- Are there other aosp branches, besides aosp-master, for which code search with cross references would be useful to you?
- From 1 to 10 (1 being the least, 10 being most), how interested would you be in access controlled code search for partner branches?
- From 1 to 10 (1 being the least, 10 being most), how important are cross references? For what languages are they important?

Develop | Iterate | **Test** | Release | Monitor

Tool	Wins	Details
Set up / maintain lab	• Simplified setup to manage your own labs that meets Google requirements • Integrate with Google's tools	• N/A
ci.android.com: presubmit / pre-merge testing	• Access builds and test results	• Android team's internal cloud-based continuous integration system
MTT: *TS deployment mechanism	• Manages retries and reporting • Eliminates need for human retries and human intervention	• N/A
AntTrail: investigate, share, compare test failures	• Store test results in Google Cloud for easy access • Will support results sharing and comparison	• Android team's internal tool for investigating and debugging test failure
Unit tests	• More deterministic quality signal upstream • Enable continuous integration	• Work in progress to identify and invest

Develop | Iterate | **Test** | Release | Monitor

Tool	Wins	Details
Integration test (e.g. camera, telephony)	• Testing camera functionality through the use of the camera app • Testing telephony across network and conditions	• N/A
UIConductor, ACTS: e2e/system test,	• Simplified QA-authored and maintained automation, with only some dev involvement	• Details covered in another talk
CrystalBall: performance test infrastructure (e.g. memory)	• N/A	• Details covered in another talk
Monkey: stability	• N/A	• N/A
App start-up framework for top100 and 3k	• AppCompat framework for validating crashes on start-up • Automated AppCompat assessment upstream	• Run in Android team's continuous integration infrastructure
CTS-V automation	• Reduce manual testing of CTS-V use cases	• Details covered in another talk

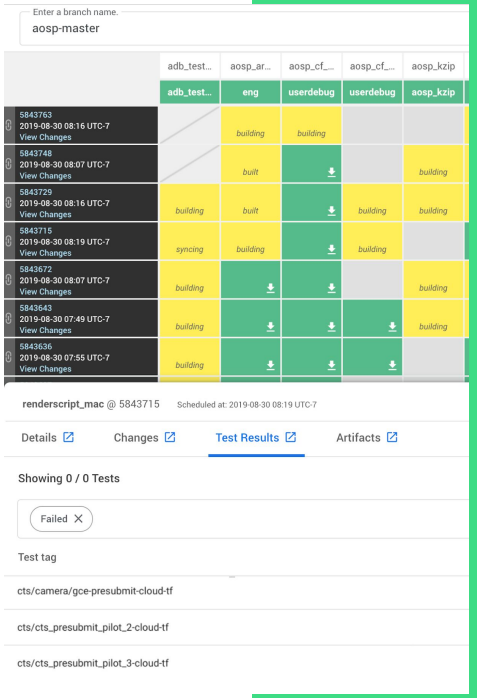
Builds and test results for partner branches on ci.android.com

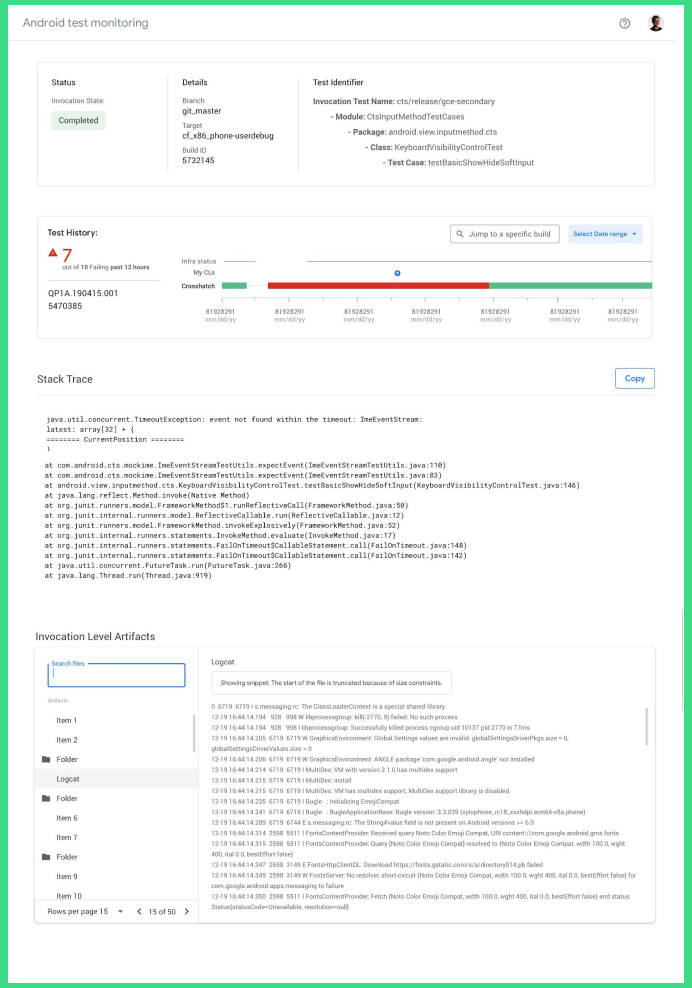
Today

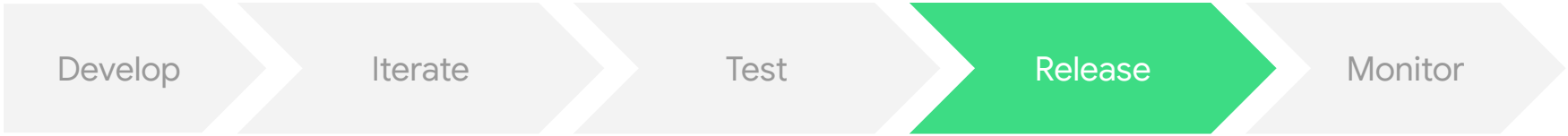
- ci.android.com shows build status and allows downloading build artifacts
- Only for aosp branches

Q1’2020

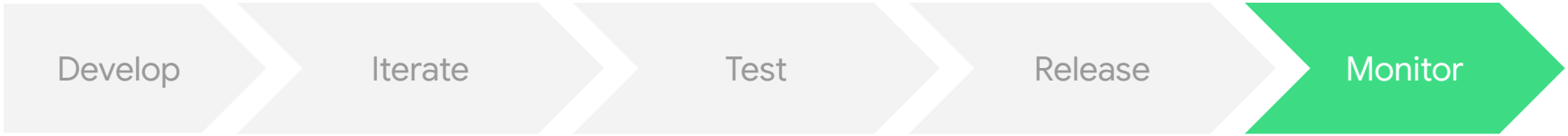
- Access controlled log in to view partner branches
- Partner branches can view build status and download build artifacts
- Partner branches can view postsubmit test results and dive into stack traces
- Test logs view is organized to easily find failed tests





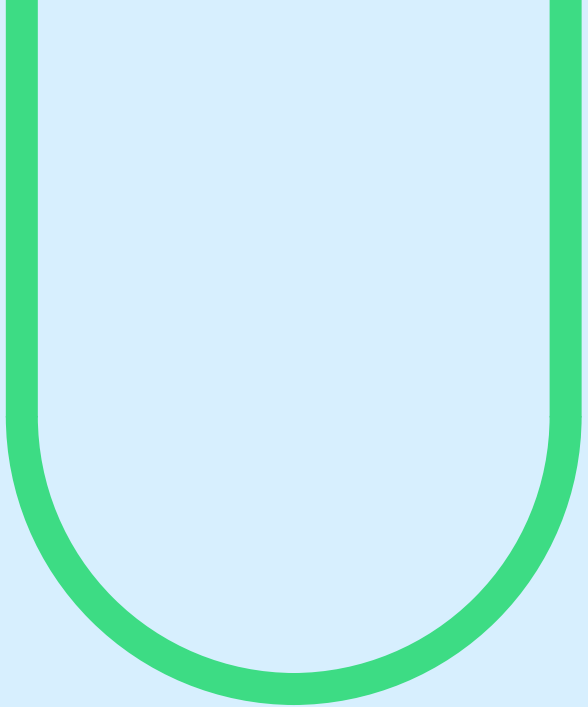


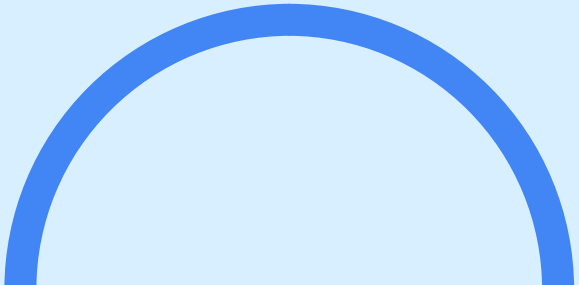
Tool	Wins	Details
Auto-merger: resolve merge conflicts	• Reduce the human time on many cases of merging across branches	• N/A
Forrest Bisection: identify culprit	• Identify the change/build that introduced the regression/bug	• UI tool + service to allow the manual bisection of builds to narrow down to the root cause



Tool	Wins	Details
Production failure debugging	• Reduce the time that the users are left in a broken state	• N/A
Growler: bug triaging / debugging / de-dupping	• Reduce the amount of time in triaging and dealing with duplicate bugs	• Auto-file and de-duplicate automated test generated failures, such as *TS, performance, etc.
Code Coverage	• Provide measurement of test coverage across Android system	• N/A

Partner interests

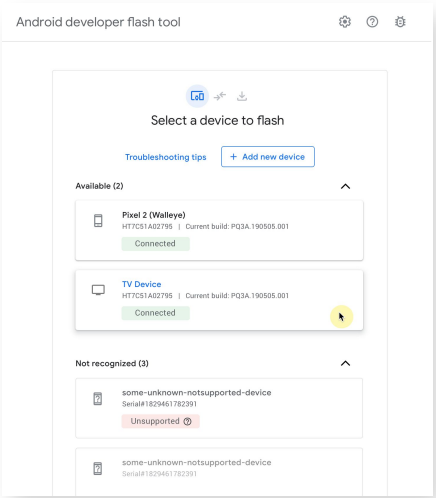




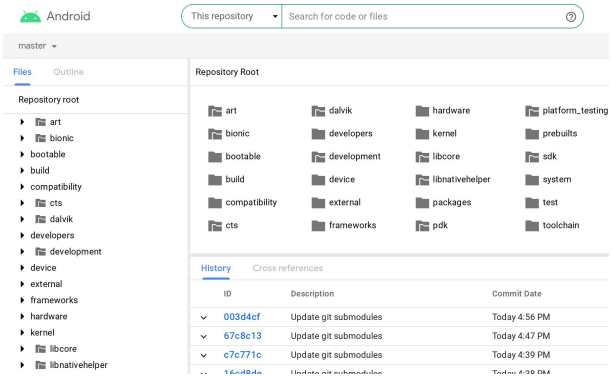
android

Recap of upcoming tools

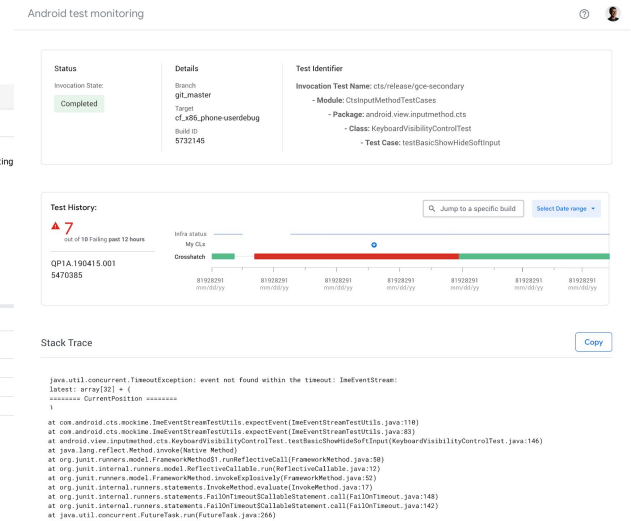
Public flashstation



Code search with cross references for aosp



Build and test results for partner branches on ci.android.com



Q/A

- What infra/tools are you most interested in? Help us understand why?
- From 1 to 10 (1 being the least, 10 being most), how interested are you in **working with + adopting Google's infra/tools** if we provide them publicly? If below 5, please help us understand why?
- From 1 to 10 (1 being the least, 10 being most), **how much effort** would you be willing to make in order to **set up an infra or tool** provided by Google?
- From 1 to 10 (1 being the least, 10 being most), how likely are you interested in **contributing back** to the community if we open source some of our infrastructure and tools? If below 5, please help us understand why?
- Any additional thoughts on what a model would best work for you?
- ***** *Stay tuned for a survey coming soon from Android to collect your feature requests* *****

Thank you for your time and attention.
Q&A and Feedback

android

Appendix

Android Development at Scale

7.4M ↗
Pre & Post-Submit Builds
(+10% YoY)

190M ↗
Lines of Code
(+15% YoY, Master)

75% ↗
CTS Presubmit
(+69pts YoY)

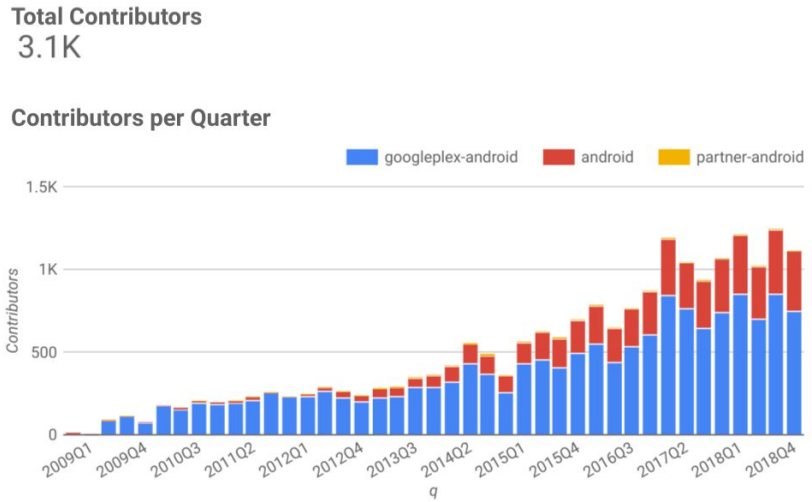
2.5M ↗
Automated Tests / month
(+90% YoY)

3% ↘
TreeHugger Flakiness
(-17pts YoY)

2hrs ↘
Postsubmit Red->Green
(-85% YoY)

39min ↘
Presubmit Build Time
(-30% YoY)

2.7hrs ↗
CL->Submission Time
(+37% YoY)



android

xTS (CTS, GTS, VTS...) Triage UI

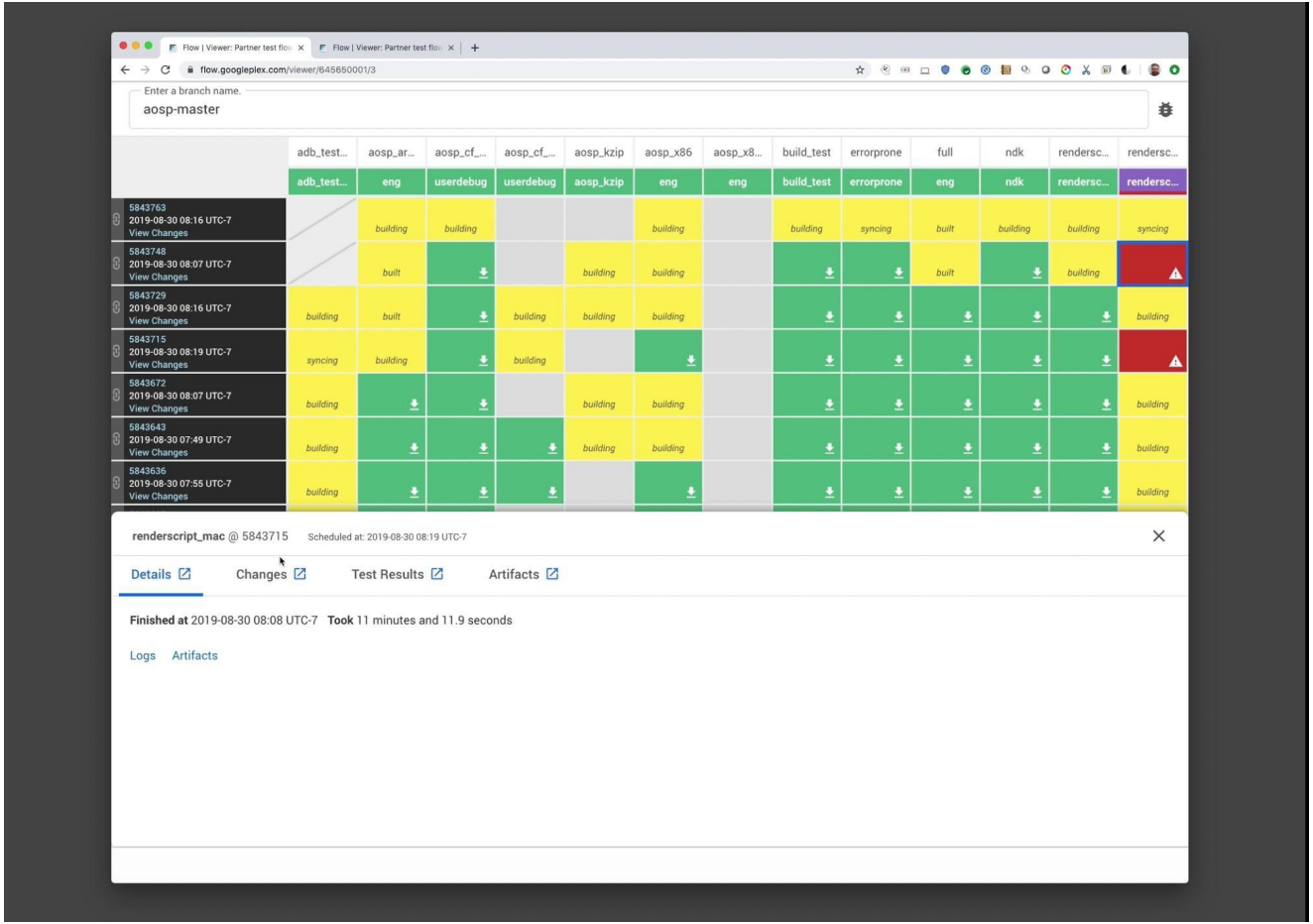
If we build a tool that helps xTS Triage

- What problems do you want it to solve?
- What should the tool look like in your mind?

Optional Test Suites

If we build a set of test suites that are optional

- Would you be willing to run them as part of your development or testing workflow?
- What are the areas of testing would you like to get help most?


flow.googleplex.com/viewer/645650001/3
Enter a branch name.
aosp-master
adb_test... aosp_ar... aosp_cf_... aosp_cf_... aosp_kzip aosp_x86 aosp_x8... build_test errorprone full ndk rendersc... rendersc...
5843763 2019-08-30 08:16 UTC-7 View Changes
5843748 2019-08-30 08:07 UTC-7 View Changes
5843729 2019-08-30 08:16 UTC-7 View Changes
5843715 2019-08-30 08:19 UTC-7 View Changes
5843672 2019-08-30 08:07 UTC-7 View Changes
5843643 2019-08-30 07:49 UTC-7 View Changes
5843636 2019-08-30 07:55 UTC-7 View Changes
renderscript_mac @ 5843715
Scheduled at: 2019-08-30 08:19 UTC-7
Details
Changes
Test Results
Artifacts
Finished at 2019-08-30 08:08 UTC-7 Took 11 minutes and 11.9 seconds
Logs
Artifacts

android

One possibility for the future

Google provides building blocks

Provide a product line of infrastructures and tools, to assist various stages of development.

Partners customize and build solutions

Choose which infrastructures and tools to adopt, and build or integrate with your own solutions