

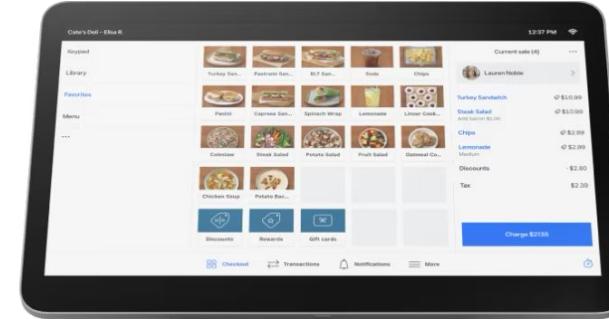
Debugging Android Devices in the Field



Chris Hayes

Android Solutions
Engineer - (He/Him)

- Enjoys: Digging into hard problems
- Previously: Square (9 years)
 - Android App Development
 - Android Build Systems
 - Android OS Platform
- Full Time remote out of Colorado, USA



AOSP Diagnostics Overview

The latest in performance monitoring and debugging logs to understand your AOSP device and fix problems



Overview

Logging



Diagnostic Tools

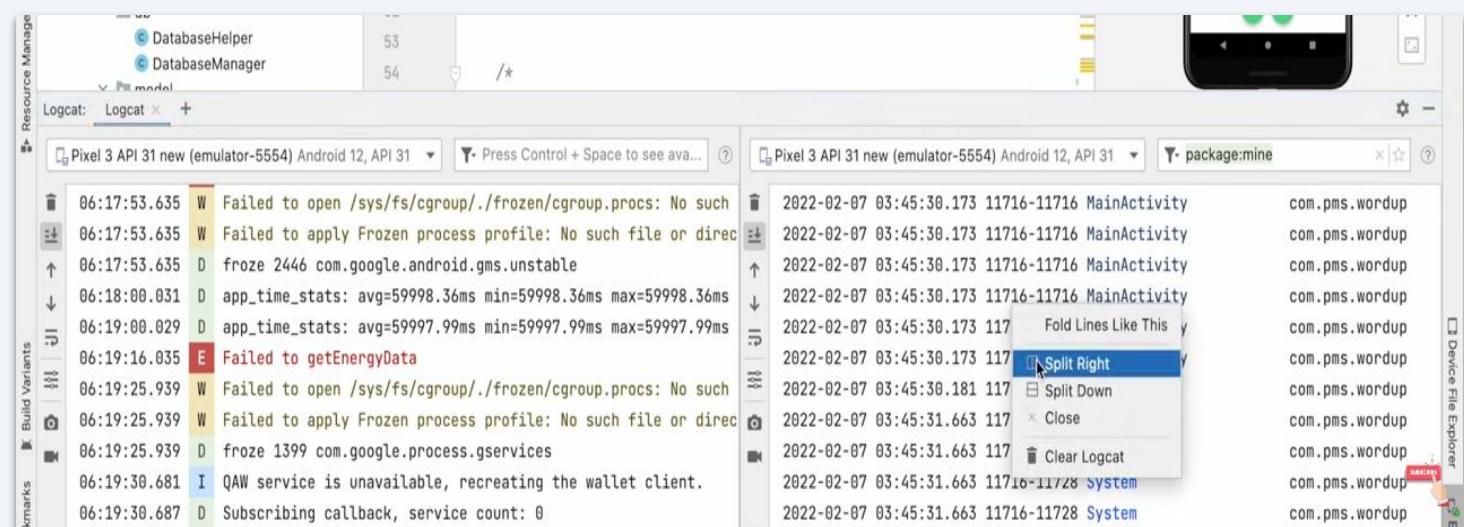
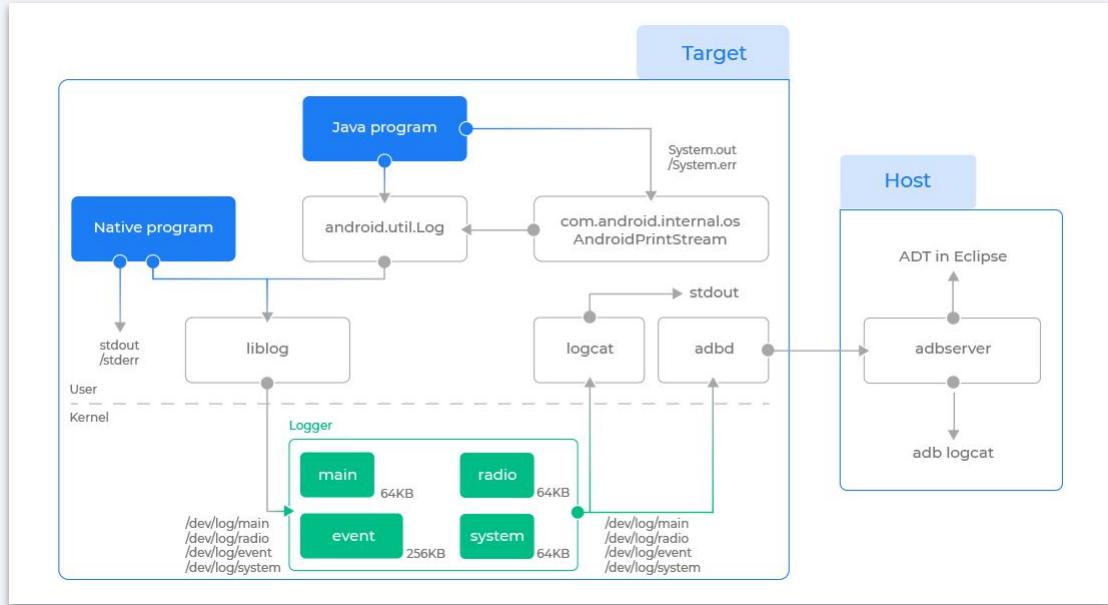


Observing Your Fleet

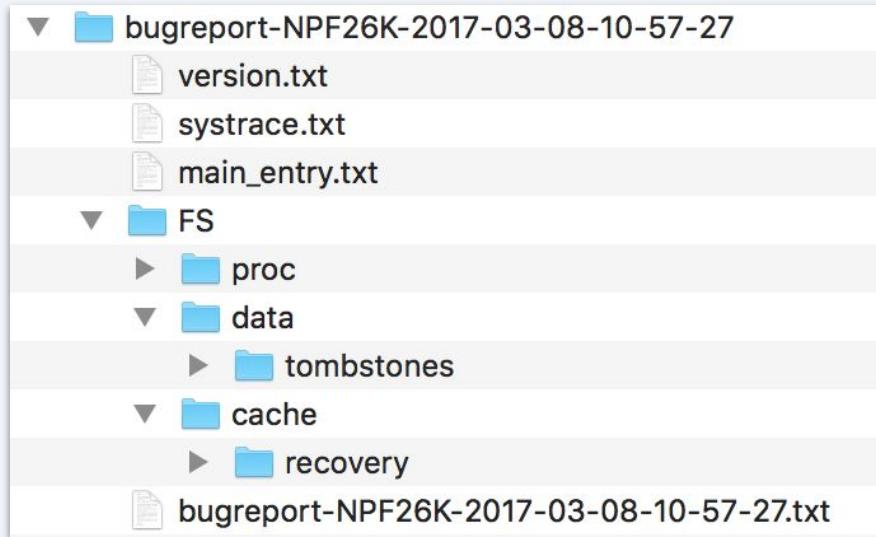


Logs for AOSP Developers

- Logcat = Circular buffers of AOSP app logs
 - main, system, crash, radio, event
 - Logcat V2 – new updates in Dolphin!
- Kmesg/dmesg = Kernel/driver logs (/proc/kmesg)
 - Great for capturing message from reboot
- **Helpful Resources**
 - Android Debugging Core Topics
 - Android Studio Logcat User Guide
 - Android Log Analysis by pCloudy
 - Logcat V2 features from googleblog
 - Logcat V2 detail overview (YouTube)



Debug Tools (log capture and analysis)



The screenshot shows a ChkBugReport log viewer interface with the following details:

- Table of contents:** A sidebar listing various log categories such as Header, Errors, Memory info, System log, Log, Spammer list, Kernel log, Event log, Battery info, AlarmManager, ActivityManager, and Processes.
- Header:** The main pane displays the dumpstate header information, which includes:
 - dumpstate: 2019-11-24 15:34:17
 - Build: MMX_TS542_SW_V03_20191101
 - Build fingerprint: 'Micromax/TS542/TS542:9/FIQ1.190601.001/125:user/release-keys'
 - Bootloader: unknown
 - Radio: JIO_3.2-00209-SIM439_GEMINI_FACK-1
 - Network: Jio 4G
 - Kernel: Linux version 4.9.112-pre6 (jdecker) (gcc version 4.9.x 20150123 (prerelease) (GCC)) #1 SMP PREEMPT Wed Nov 6 14:32:05 CST 2019
 - Command line: sched_enable_hrtsl console=ttyMSM0,115200,n8 androidboot.console=ttyMSM0 androidboot.hardware=qcom msm_rtb.filter=0x237 ehci-hcd
 - Uptime: up 0 weeks, 0 days, 0 hours, 0 minutes
 - Bugreport format version: 2.0
 - Dumpstate info: id=1 pid=3669 dry_run=0 args=/system/bin/dumpstate -d -p -B -z -o /data/user_de/0/com.android.shell/files/bugreports/bugreport
 - Plugin crashed while loading data: com.sonyericsson.chkbugreport.plugins.stacktrace.StackTracePlugin

- ADB – Android Debug Bridge
 - Connect your host to device, get logs from logcat
- Bug Reports – Snapshot of logs for troubleshooting
 - Dumpsys – System services
 - Dumpstate – Error logs
 - Logcat – System messages
- DropBoxManager – More targeted reports
 - Data specifically from apps that crash

Helpful Resources

- Android Debugging Core Topics
- Android Studio Bug Report User Guide
- Android Bug Report Videos (YouTube)
- A helpful viewer: ChkBugReport

LNAV

- Feature rich log navigator
- SQL query language built in
- Histogram view
- Multi-log interpolation
 - Merge logcat and kernel logs
- Supports custom log formats
- Syntax highlighting
- Custom regex highlighting
- Pretty-print structured data

• Helpful Resources

- [Inav features](#)
- [Android Log Format](#)

The screenshot shows the LNAV application interface. The left pane, titled 'LOG — Inav - Inav/inav/docs/tutorials/playground/logs — 114x45', displays a log file with multiple entries. The log entries are color-coded by source and type, with some lines highlighted in blue. The right pane, titled 'PRETTY — Inav — 114x35', shows a single XML configuration file with various parameters and their values, also color-coded for readability.

Left Pane (LOG):

```
2022-09-07T14:09:34 PDT
LOG | 2022-01-13T10:24:08.000)access_log)access_log.gz[2,473]192.0.2.122
192.0.2.122 - combatcarl@example.com [13/Jan/2022:10:24:08 +0000] "GET /index.html HTTP/1.0" 200 7149 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/98.0.4758.102 Safari/537.36"
192.0.2.42 - - [13/Jan/2022:10:24:09 +0000] "GET dolor/amet/eiusmod HTTP/1.0" 403 1263 "-" "Roku4640X/DVP-7.70 (192.0.2.55"
192.0.2.55 - combatcarl@example.com [13/Jan/2022:10:24:11 +0000] "PUT aliqua/aliqua/et HTTP/1.0" 404 3233 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/98.0.4758.102 Safari/537.36"
Jan 13 10:24:43 frontend3 server[123]: Received packet from 192.0.2.42
Jan 13 10:24:45 frontend3 worker[61457]: Successfully started helper
Jan 13 10:24:46 frontend3 server[124]: Received packet from 192.0.2.122
Jan 13 10:24:47 frontend3 server[121]: Received packet from 192.0.2.44
Jan 13 10:24:49 frontend3 server[124]: Received packet from 192.0.2.42
192.0.2.42 - combatcarl@example.com [13/Jan/2022:10:27:31 +0000] "GET consectetur/et/et/consectetur/sed/et elit HTTP/1.0" 200 1140 "-" "Roku4640X/DVP-7.70 (192.0.2.42"
192.0.2.42 - - [13/Jan/2022:10:27:33 +0000] "GET sed/amet/inciduntd HTTP/1.0" 200 1140 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/98.0.4758.102 Safari/537.36"
192.0.2.42 - combatcarl@example.com [13/Jan/2022:10:27:34 +0000] "GET dolor/sed/Lorem HTTP/1.0" 200 5703 "http://192.0.2.122" - - [13/Jan/2022:10:27:35 +0000] "GET dolor/amet/eiusmod HTTP/1.1" 200 1669 "-" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/98.0.4758.102 Safari/537.36"
192.0.2.122 - combatcarl@example.com [13/Jan/2022:10:27:36 +0000] "GET amet/eiusmod/magna/ipsum/amet HTTP/1.0" 200 6300
Jan 13 10:28:09 frontend3 server[123]: Successfully started helper
Jan 13 10:28:10 frontend3 server[121]: Received packet from 192.0.2.42
Jan 13 10:28:11 frontend3 server[124]: Received packet from 192.0.2.42
Jan 13 10:28:13 frontend3 server[123]: Handling request 336d2eec-6c24-4d4b-9517-0a408d19ab4d
192.0.2.3 - bob@example.com [13/Jan/2022:10:35:15 +0000] "GET do/sit/ipsum HTTP/1.1" 403 243 "-" "Apache/Httpd/2.4.41"
192.0.2.55 - combatcarl@example.com [13/Jan/2022:10:35:16 +0000] "PUT dolore/labore/sed HTTP/1.0" 200 2035 "http://192.0.2.42" - - [13/Jan/2022:10:35:17 +0000] "GET /obj/1235?foo=bar HTTP/1.0" 403 7635 "-" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/98.0.4758.102 Safari/537.36"
192.0.2.42 - combatcarl@example.com [13/Jan/2022:10:35:18 +0000] "GET sit/ut/ut/dolore/Lorem HTTP/1.0" 404 6300
Jan 13 10:35:52 frontend3 worker[61457]: Handling request b8cf7d43-da8c-465f-bc34-61f6dcdb0bbc
Jan 13 10:35:53 frontend3 server[123]: Received packet from 192.0.2.42
Jan 13 10:35:55 frontend3 server[123]: Received packet from 192.0.2.122
Jan 13 10:35:56 frontend3 worker[61457]: Reading from device: /dev/hda
Jan 13 10:35:58 frontend3 server[123]: Handling request 336d2eec-6c24-4d4b-9517-0a408d19ab4d
Jan 13 10:35:59 frontend3 worker[61456]: Handling request b8cf7d43-da8c-465f-bc34-61f6dcdb0bbc
192.0.2.122 - - [13/Jan/2022:10:48:37 +0000] "GET sed/adipiscing/dolor/labore/magna/ipsum/do/elit HTTP/1.1" 404
192.0.2.42 - bob@example.com [13/Jan/2022:10:48:39 +0000] "GET incididunt/labora/aliqua/Lorem/magna HTTP/1.1" 200
192.0.2.42 - bob@example.com [13/Jan/2022:10:48:40 +0000] "GET eiusmod HTTP/1.1" 200
192.0.2.122 - combatcarl@example.com [13/Jan/2022:10:48:42 +0000] "GET /index.html HTTP/1.0" 200
192.0.2.42 - combatcarl@example.com [13/Jan/2022:10:48:43 +0000] "GET elit/ipsum/elit/sed/et elit HTTP/1.1" 404
192.0.2.122 - - [13/Jan/2022:10:48:44 +0000] "GET /index.html HTTP/1.0" 200
192.0.2.3 - - [13/Jan/2022:10:48:46 +0000] "GET /index.html HTTP/1.0" 200
Jan 13 10:49:18 frontend3 server[123]: Received packet from 192.0.2.42
Jan 13 10:49:19 frontend3 server[124]: Received packet from 192.0.2.42
Jan 13 10:49:21 frontend3 server[124]: Received packet from 192.0.2.42
Jan 13 10:49:22 frontend3 server[123]: Handling request b8cf7d43-da8c-465f-bc34-61f6dcdb0bbc
```

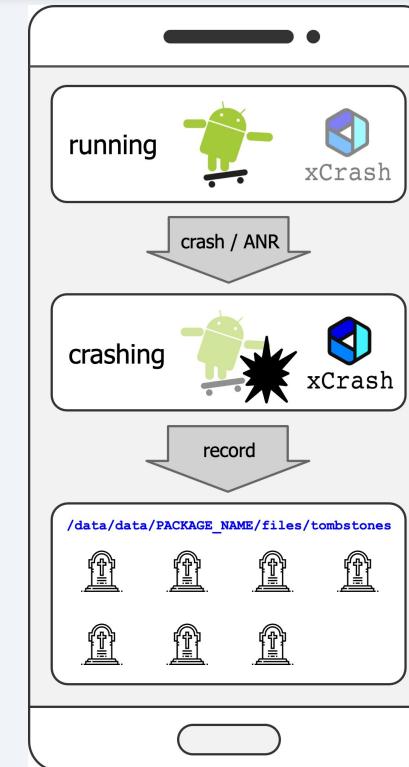
Right Pane (PRETTY):

```
<?xml version="1.0"?>
<response>
  <locale>en-US</locale>
  <requestid>ipInfo</requestid>
  <value id="ipv4Gateway" actions="enabled">10.133.235.253 (unknown)</value>
  <value id="ipv6Gateway" actions="enabled"/>
  <value id="ipv4Enabled" actions="enabled">true</value>
  <value id="ipv6Enabled" actions="enabled">true</value>
  <value id="nmap" actions="enabled">nic1</value>
  <value id="v4config" actions="enabled">
    <value id="defaultGateway" actions="enabled"><0.0.0.0 (unknown)</value>
    <value id="updateable" actions="enabled">true</value>
    <value id="prefix" actions="enabled">22</value>
    <value id="mode" actions="enabled">dhcp</value>
    <value id="address" actions="enabled">10.133.234.110 (unknown)</value>
    <value id="interface" actions="enabled">nic1</value>
  </value>
  <value id="v6config" actions="enabled">
    <value id="defaultGateway" actions="enabled">fe80::214:f609:19f7:6bf1 (unknown)</value>
    <value id="updateable" actions="enabled">true</value>
    <value id="interface" actions="enabled">nic1</value>
    <value id="dhcp" actions="enabled">false</value>
    <value id="autocom" actions="enabled">false</value>
    <value id="addresses" actions="enabled">
      <value id="origin" actions="enabled">other</value>
      <value id="status" actions="enabled">preferred</value>
      <value id="prefix" actions="enabled">64</value>
      <value id="address" actions="enabled">fe80::250:56ff:fea:5abf (unknown)</value>
    </value>
  </value>
</response>
```

Android Crash Data from the Logs and Tools

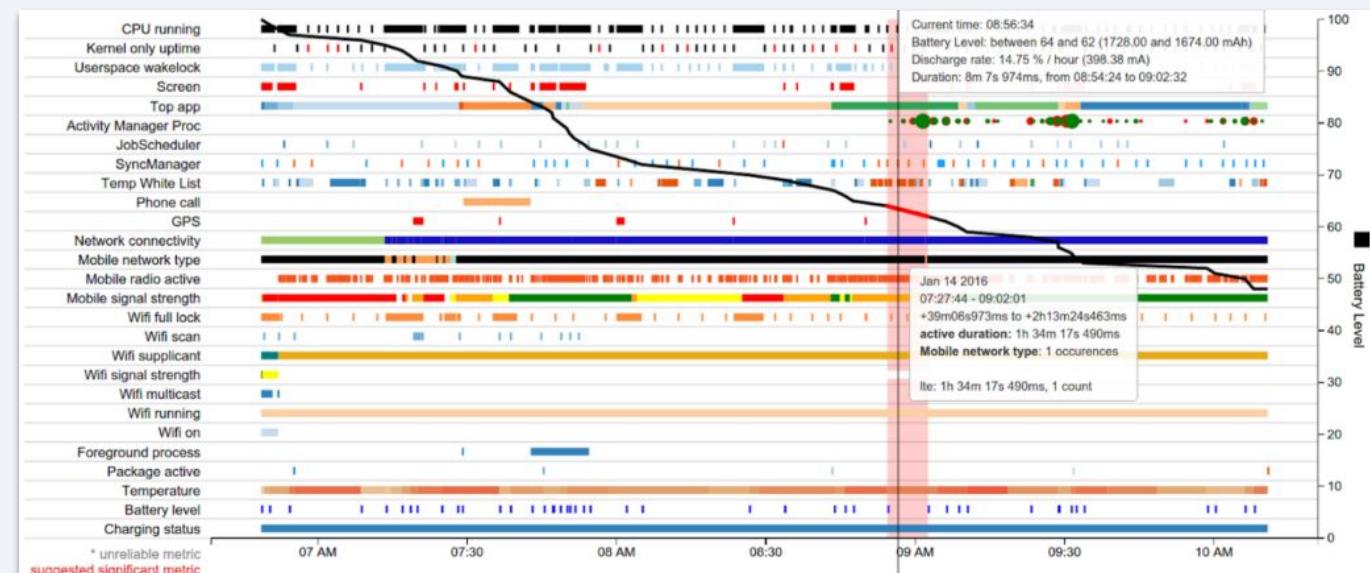
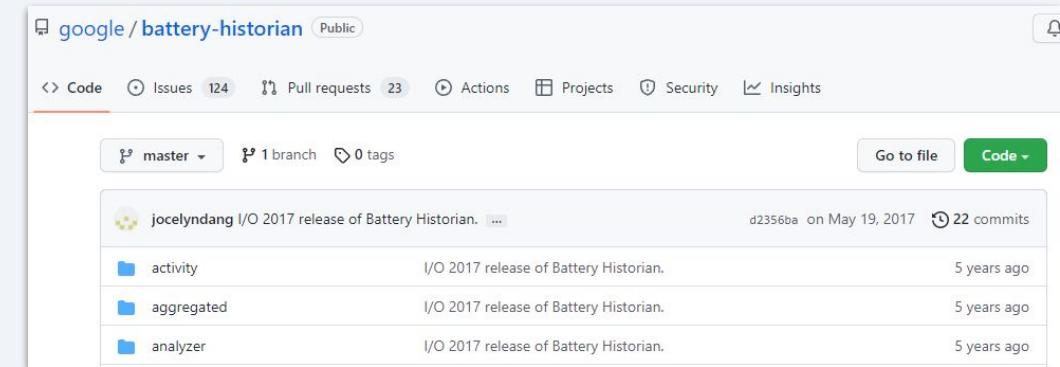
- Tombstones - More detail than logcat from crash dump
 - `/data/tombstones/<file>`
 - Stack traces of all processes, memory map, open files
- ANRs – Application Not Responding (5 seconds)
- Kernel Oops – Serious, possibly fatal kernel errors
 - Found in logcat
- WTFs – an assert triggered by logcat
- Java exceptions
- SELinux policy violations
 - In audit.d logs
- **Helpful Resources**
 - Debugging Native Crashes in Android Apps

```
< Your System ate a SPARC! Gah! >
-----
\ \ ^__^
  (xx)\  -----
  ( )\  |
    ||----w |
    |
sshd (pid 19569): Protection id trap (code 27)
```



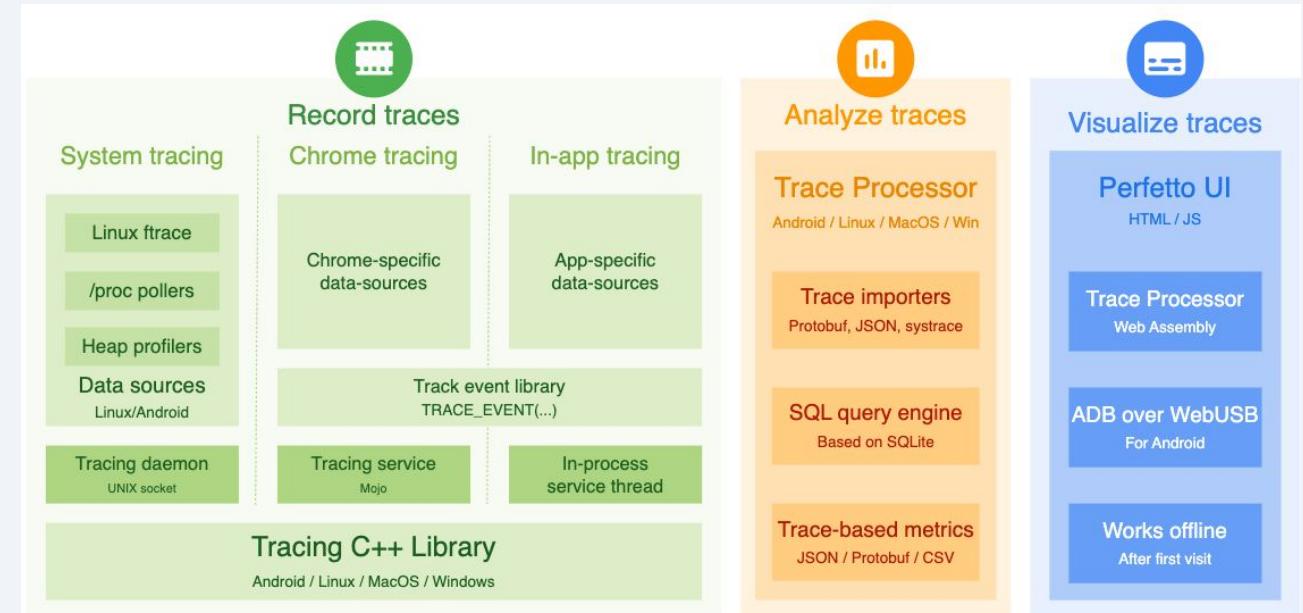
BatteryStats & Battery Historian

- BatteryStats – collects battery data on device
 - Which processes are drawing current?
 - When are they doing it?
 - Adb can get logs
- Battery Historian – a viewer for BatteryStats
 - Can consume a bug report (logs are in it)
 - Can show System Stats and App Stats
- **Helpful Resources**
 - [Battery Historian GitHub](#)
 - [Battery Historian Video \(YouTube\)](#)
 - [Deeper Dive on Battery Historian \(YouTube\)](#)



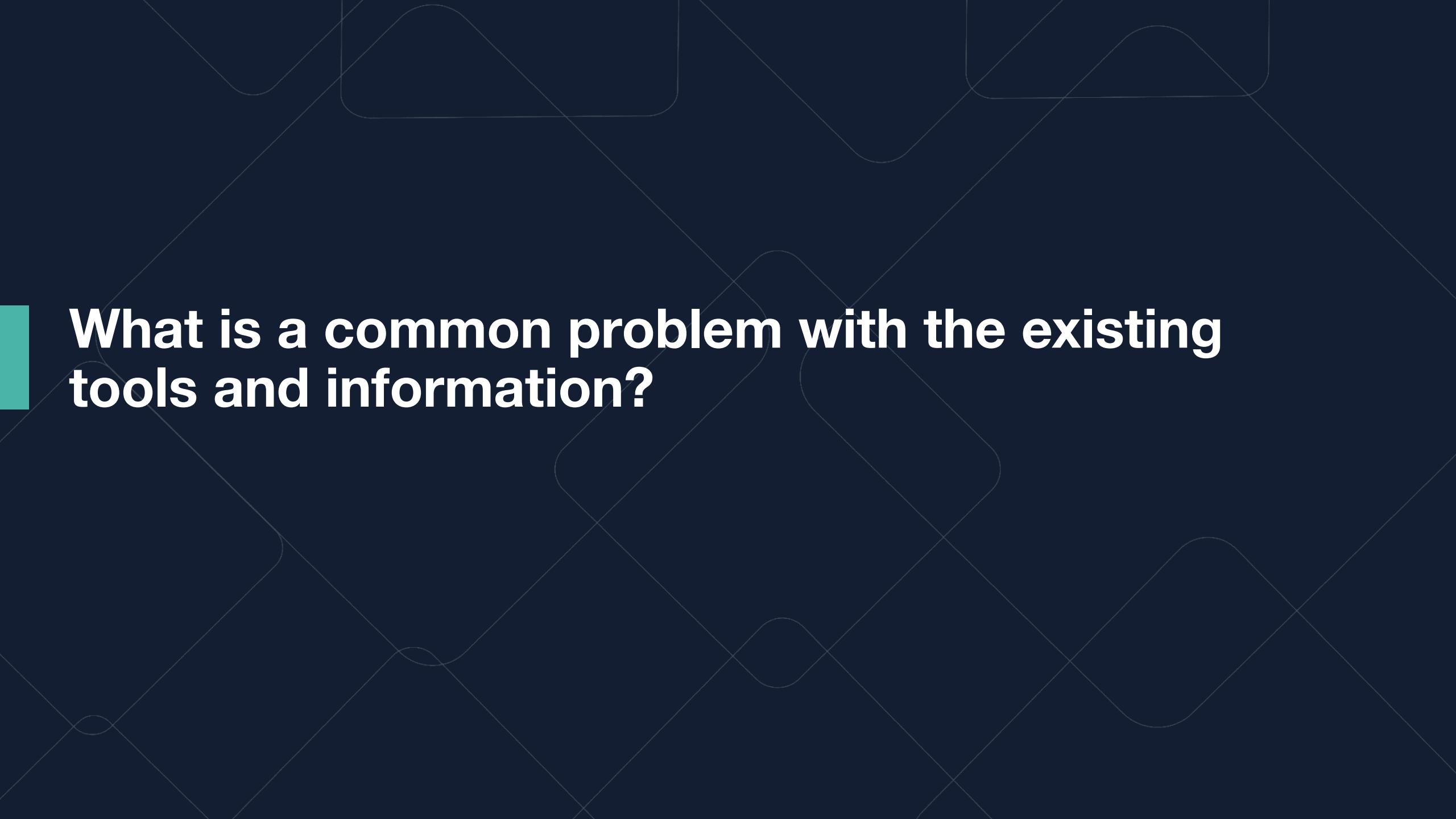
Performance Tracing & Monitoring

- Perfetto
 - Full system tracing framework and analysis tools.
 - Capture high frequency ftrace data: scheduling activity, task switching latency, CPU frequency and much more
- Leak Canary
 - A memory leak detection library for Android.
 - Built in heap analyzer
 - Allows for uploading heap analysis to third party services



• Helpful Resources

- [Perfetto GitHub](#)
- [Perfetto Docs](#)
- [Leak Canary Getting Started](#)



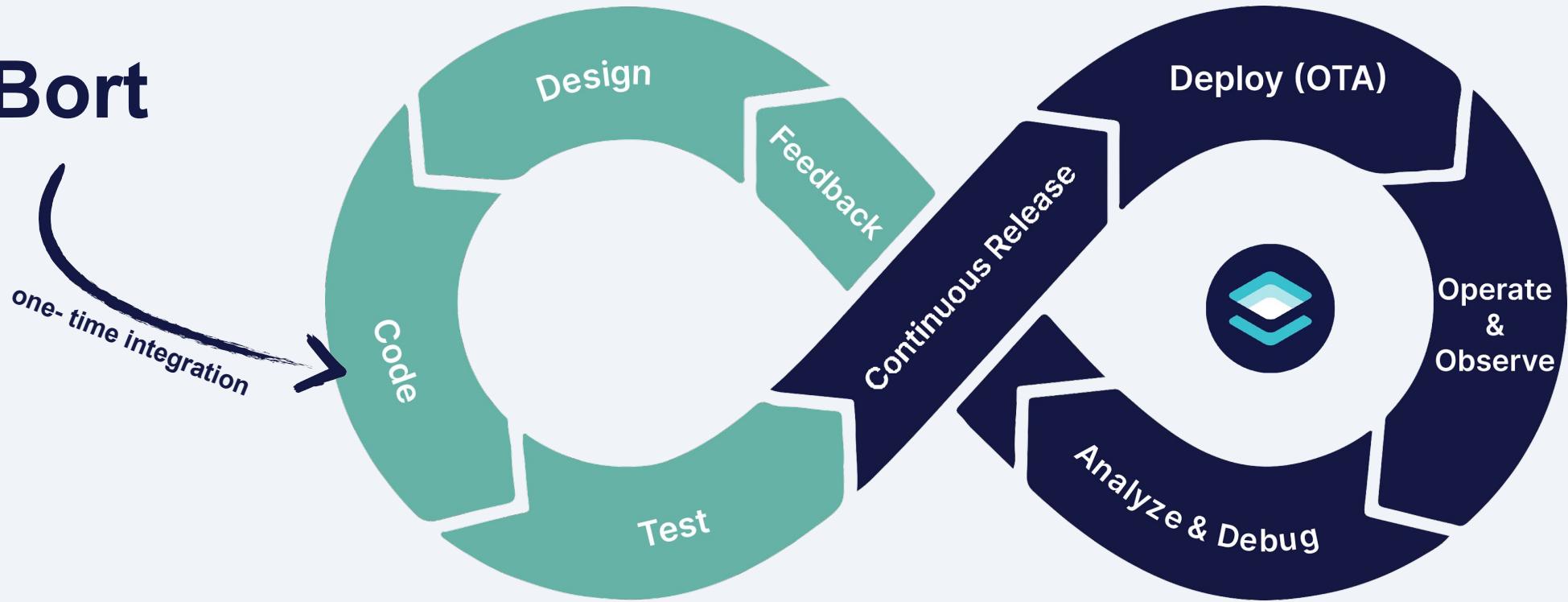
What is a common problem with the existing tools and information?



Memfault

Applying DevOps Thinking

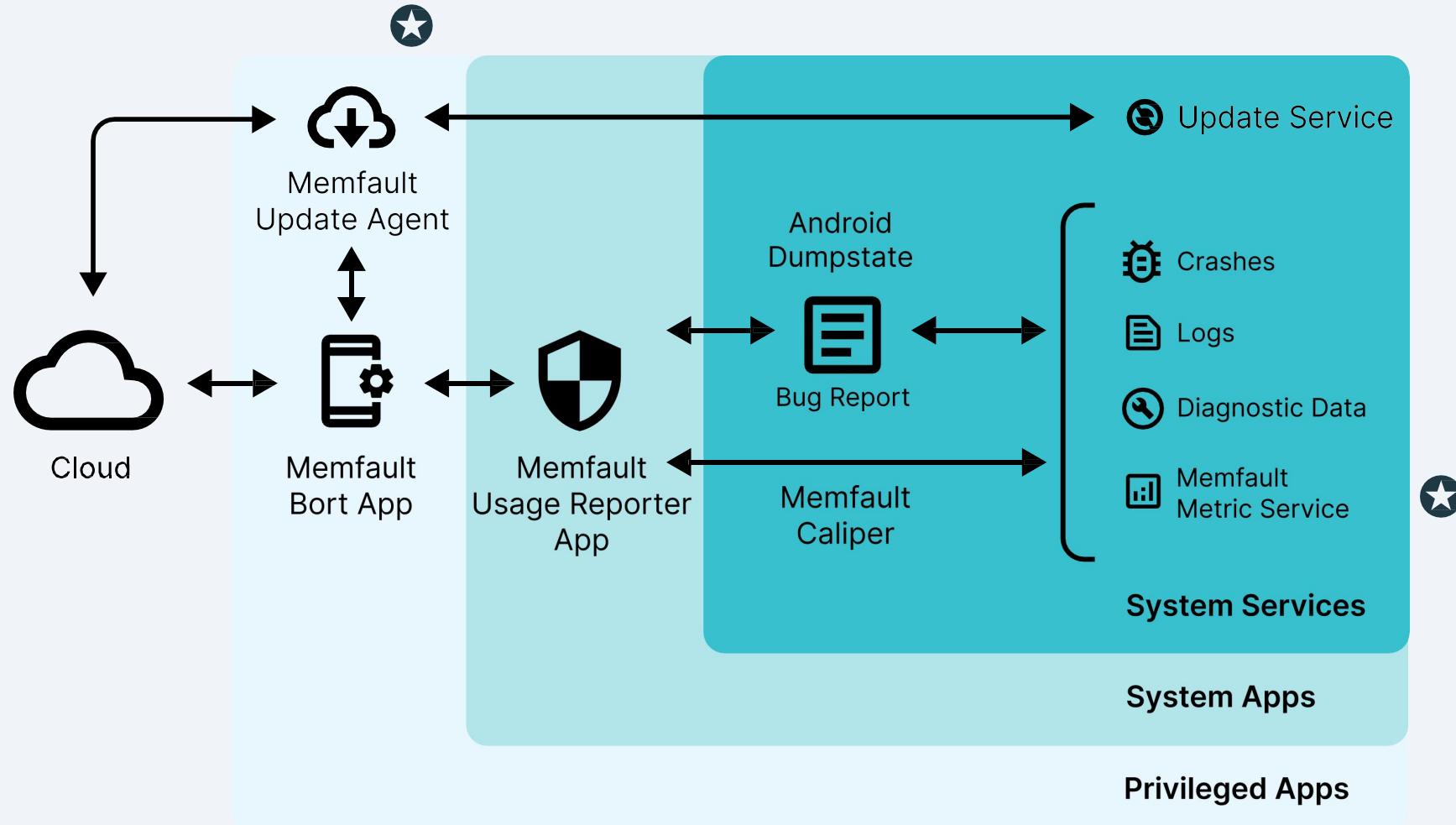
Bort



Development Process

Fleet Management & Observability

Inside Bort, the Memfault AOSP SDK



Observing Your Fleet with Memfault

The image displays three side-by-side screenshots of the Memfault web application interface, illustrating its features for fleet management and incident response.

- Screenshot 1: Fleet Overview**
This screenshot shows the main dashboard for the "Smart Fridge" project. It includes:
 - Active Devices:** A line chart showing device count over time (Monthly, Weekly, Daily). The count has increased from ~7 to ~28 over the last 24 hours.
 - Software Versions:** Two charts showing software version distribution over 2 months and 24 hours. A pie chart indicates 88% android-build, 4% eng-build, and 4% eng.dock.
 - Received Traces:** A bar chart showing trace counts for the last 24 hours, with the highest count being 200 for the "Oops at v4l_enum_fmt" issue.
 - Reboots:** A stacked bar chart showing reboot counts by reason (userrequested, shutdown, battery) over the last 24 hours.
 - Newest Issues:** A list of recent issues including "Oops at v4l_enum_fmt", "CANNOT LINK EXECUTABLE", "Native", "ANR", "syscall", "Native", "signal 6 (SIGABRT)", "code -6 (SIGKILL)", and "Exception".
- Screenshot 2: Device Details**
This screenshot shows the details for a specific device, "MFLTPV0001". It includes:
 - Serial Number:** MFLTPV0001
 - Nickname:** not set
 - Software Version:** None
 - First Seen:** Mar 11, 2022 9:37 AM
 - Cohort:** Internal
 - Links:** A link to the device's ShID: MFLTX0009.
 - Timeline:** A detailed timeline showing activity from March 23 to 24, 2022, across various metrics like Traces, Attributes, Log Files, Reboots, Bug Reports, and Timeline.
- Screenshot 3: Issue Timeline**
This screenshot shows the timeline for a specific issue, "signal 6 (SIGABRT), code -6 (SIGKILL)". It includes:
 - Details:** Information about the device (MFLTDV0008), cohort (Production), software (eng.root.20201008.150503), and hardware (dvt).
 - Logs:** A log entry for "/vendor/bin/hw/android.hardware.bluetooth@1.0-service-qti [11876]" showing a stack trace of 13 lines of Java and C++ code.



Live Demo

Learn more about AOSP Tools and Memfault

On-Demand Webinar

How to Debug, Update & Monitor Embedded Android Devices

Recorded: Thursday, October 28, 2021



On-Demand Webinar

Managing Android Devices at Scale with Memfault AOSP SDK v4.0

Recorded: Thursday, March 24, 2022



Speaker



Ryan Case
Director of Engineering,
Memfault

[Link to Webinar](#)

Speaker



Heiko Behrens
Head of Product, Memfault

[Link to Webinar](#)

The AOSP and AAOS Meetup

Meetup Group

Led by Chris Simmonds

Based out of the UK

Remote Friendly

Talks all about AOSP and AAOS!



Thank You!

- memfault.com/android
- twitter.com/memfault
- linkedin.com/company/memfault
we're hiring!



Memfault



Chris Hayes

Android Solutions, Memfault