Keeping your Android application running when the device wants to sleep

Darryn Campbell, Application architect
Zebra Technologies

December 2018

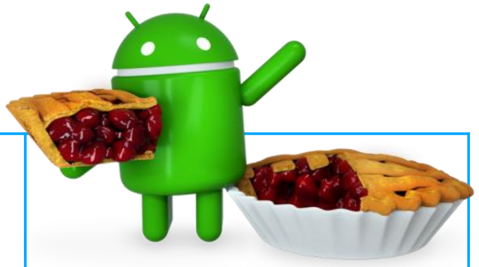
Introduction

Why do applications need to keep running in the background?

- Darryn Campbell
  - Background in Android in the Enterprise
  - Lessons learned applicable to all Android apps
- Sometimes applications find themselves fighting against the grain
  - Struggling to adhere to power management best practice… You are not alone!

Current state:

Changes to background behavior over time

					
Running in the background	**Job Scheduler**	**Doze mode**	**Doze “on the go”**	**Background restrictions**	**Machine learning for intelligent restrictions**

Each new iteration of Android introduces new considerations for applications’ background behaviour

Current state:

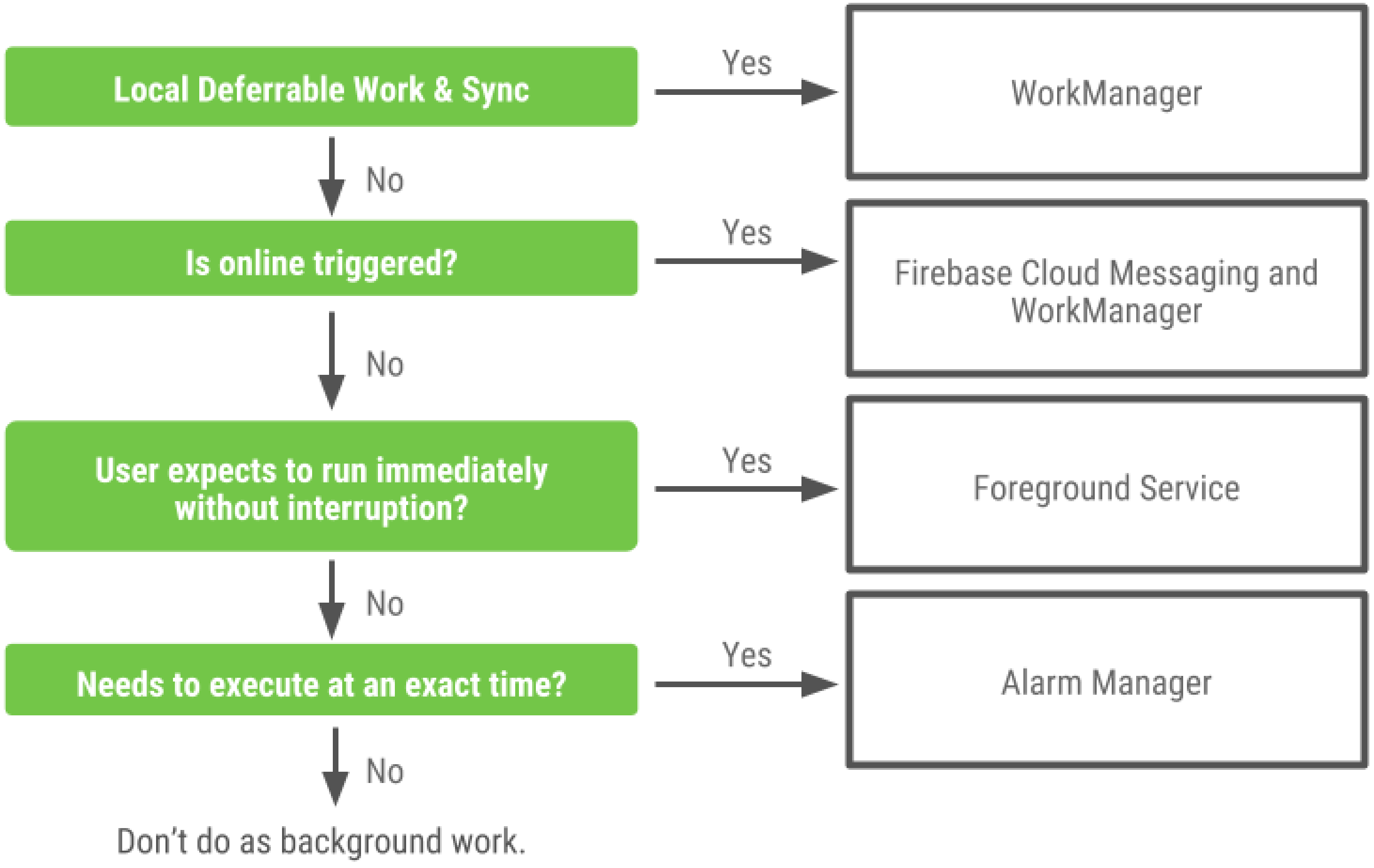
Available advice from Google

- Google publish documentation in each new release
  - Lollipop, Marshmallow, Nougat, Oreo, Pie
    - Includes samples, behaviour changes, API changes & other info
      - Doze mode explained for Marshmallow and the changes made in Nougat. Acceptable whitelisting use cases for doze mode.
      - Oreo background restrictions are explained in detail here and here.
    - Blogs such as this one on background execution and this one on power management under Pie.
  - YouTube videos on their Developers channel:
    - E.g. Background changes in Oreo, what’s new in Oreo and IO2018’s video on battery drain, Android Summit 2018’s background app changes for P and using the WorkManager.
- The advice available may not be suitable for everybody:
  - I don’t have the time to understand every nuance of power behaviour

Current state: Advice available

Latest recommendation involves the WorkManager API

I need to run a task in background, how should I do it?



Local Deferrable Work & Sync — Yes → WorkManager

No

Is online triggered? — Yes → Firebase Cloud Messaging and WorkManager

No

User expects to run immediately without interruption? — Yes → Foreground Service

No

Needs to execute at an exact time? — Yes → Alarm Manager

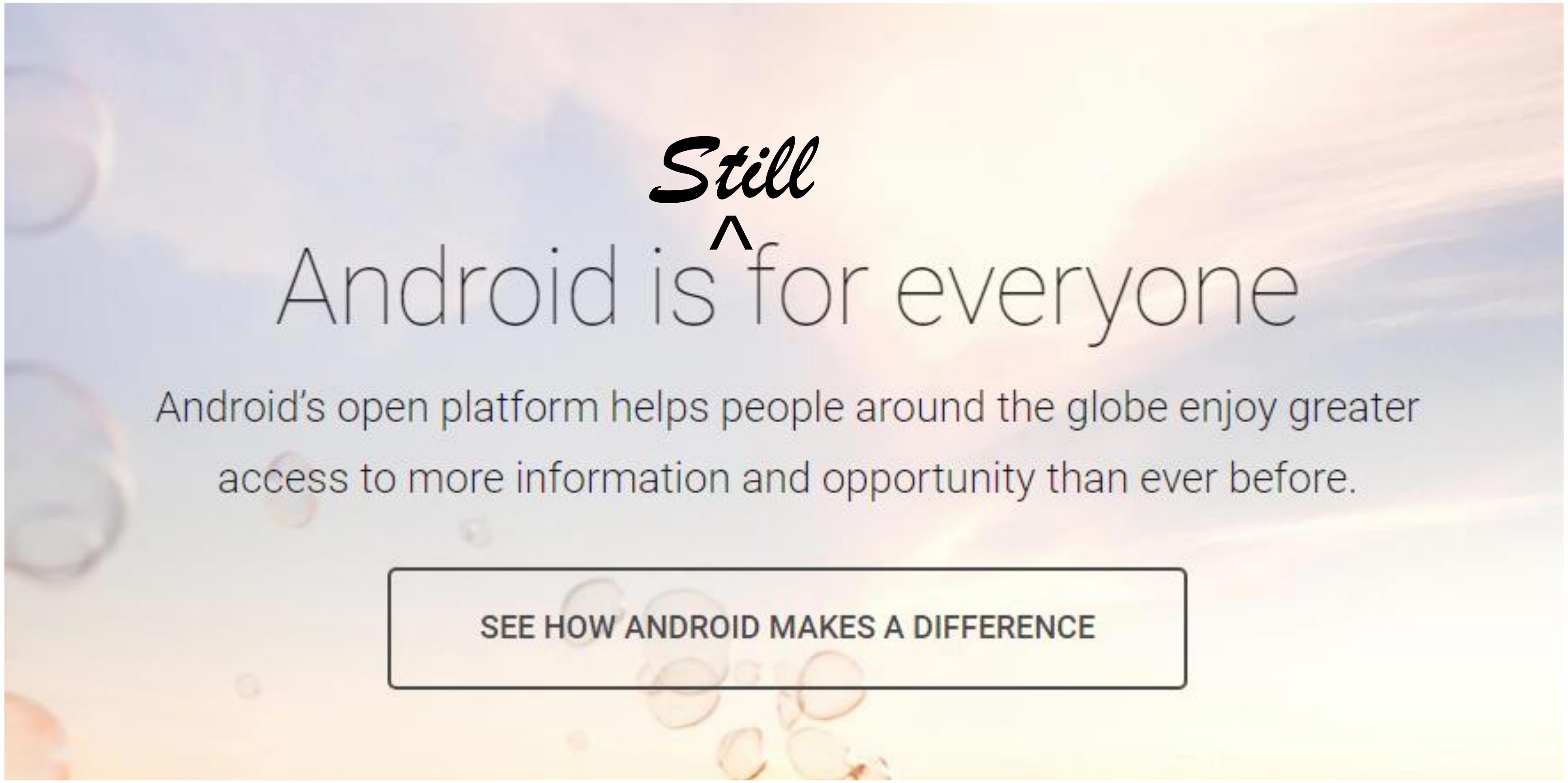
No

Don’t do as background work.

Source: Android Developers Blog

Customer case studies covering 3 industries

More innovative “solutions” to working in the background



Still
^

Android is for everyone

Android’s open platform helps people around the globe enjoy greater access to more information and opportunity than ever before.

SEE HOW ANDROID MAKES A DIFFERENCE

Customer 1

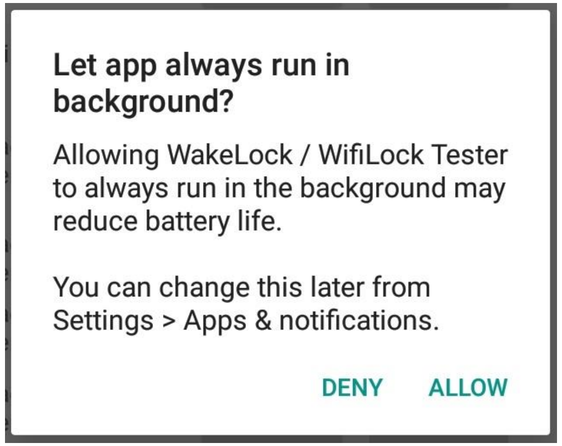
Warehouse device tracker: Problem statement

- Think “Find My Device” but within the 4 walls
  - No external network access:
    - No Google accounts or FCM.
    - No WAN connection
  - Find devices → Device chirps loudly
  - User finds device → Chirping stops

Customer 1

Warehouse device tracker: Initial concept

- Device needs constant connection to communicate location every 5 minutes
  - Therefore: Whitelist the app to avoid doze
  - Whitelisting required the user to agree to disable battery optimization manually
    - "What if the user provisioning the device clicks 'Deny'?"



Let app always run in background?

Allowing WakeLock / WifiLock Tester to always run in the background may reduce battery life.

You can change this later from Settings > Apps & notifications.

- Proof of concept saw battery consumption increase by 7%

¯_(ツ)_/¯

Customer 1

Warehouse device tracker: Final Concept

r/ProgrammerHumor

READMEs are just suggestions anyways

u/Jakeob28 • 11h • i.redd.it

I Am Devloper
@iamdevloper

Remember, a few hours of trial and error can save you several minutes of looking at the README.

2:11 AM · 07 Nov 18

- This was a perfect example where following Google’s guidance was the right thing to do
  - Move to Work Manager and schedule a periodic job to report device location
  - Use a foreground service to circumvent doze mode for the device chirping logic
- **Lessons learned:**
  - Discuss requirements with the client – the 5 minute reporting interval was being treated as an exact interval
  - Spend 30 minutes reading articles on power management rather than hours debugging why your app is non-responsive or battery drain is excessive

Customer 2

Healthcare task assignment system: Problem statement

- Within a hospital
  - – Assign tasks to nurses / doctors / staff
  - – Some messages very time sensitive
  - – Emergency tasks could (literally) be a matter of life or death
- Customer not comfortable with Firebase Cloud Messages
  - – FCM very reliable but outages do occur
  - – Round-trip-time to external FCM unacceptable

Customer 2

Healthcare task assignment system: Initial concept

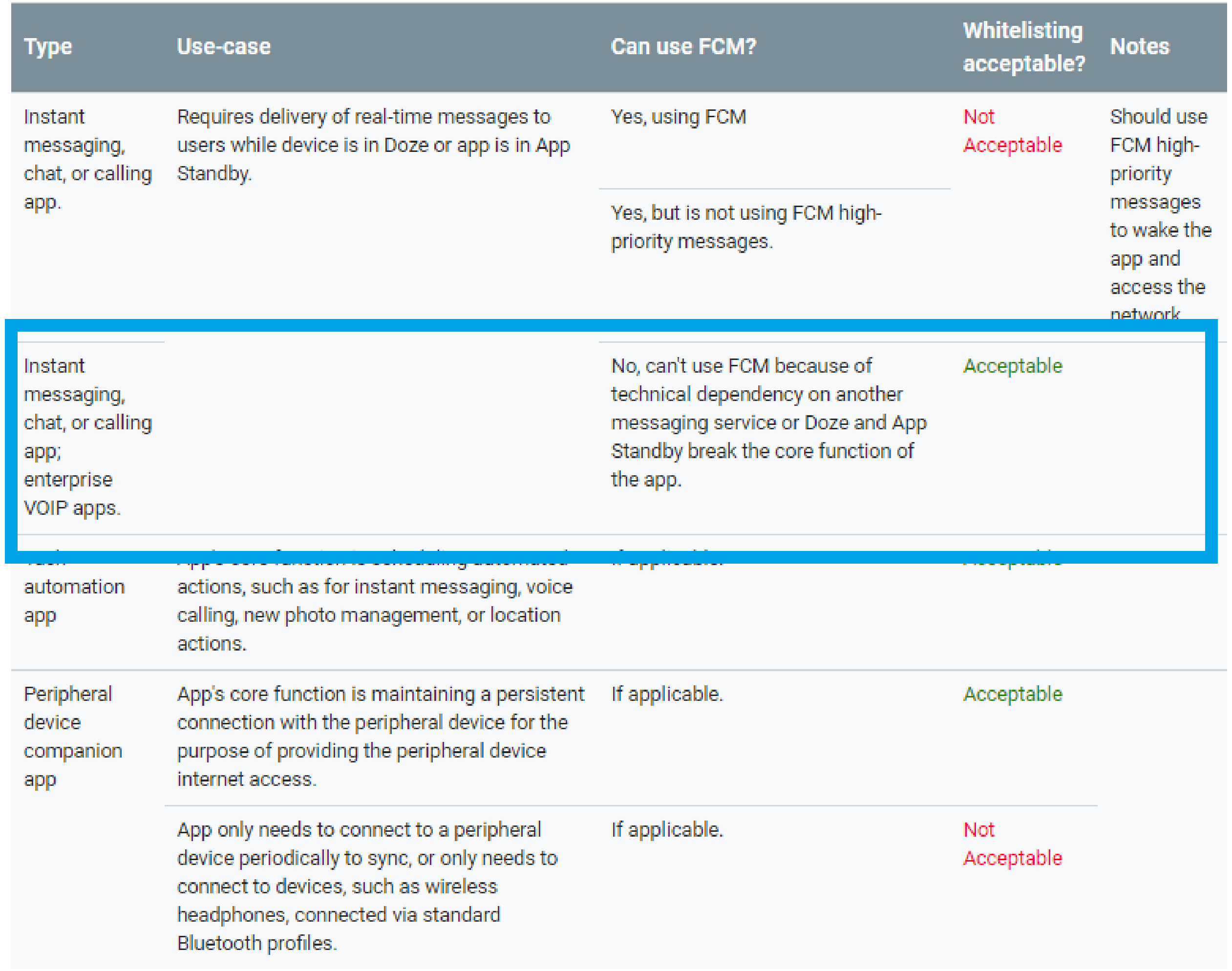
- System deployed and working well under Lollipop
  - – Self-hosted push-messaging system based on MQTT
  - – Uptime can be guaranteed by hospital IT administrator
  - – Any changes to the software / hardware are seen as a risk

- Customer needs to move to Marshmallow
  - – Receives latest security patches (at the time)
  - – Deployed software stops working when device enters doze mode
  - – Customer was using a foreground service as a workaround

Customer 2

Healthcare task assignment system: “Final” concept

- Use case falls under acceptable Doze mode whitelisting use cases:



Type	Use-case	Can use FCM?	Whitelisting acceptable?	Notes
Instant messaging, chat, or calling app.	Requires delivery of real-time messages to users while device is in Doze or app is in App Standby.	Yes, using FCM	Not Acceptable	Should use FCM high-priority messages to wake the app and access the network
		Yes, but is not using FCM high-priority messages.		
Instant messaging, chat, or calling app; enterprise VOIP apps.		No, can't use FCM because of technical dependency on another messaging service or Doze and App Standby break the core function of the app.	Acceptable	
Task automation app	App's core function is scheduling automated actions, such as for instant messaging, voice calling, new photo management, or location actions.	If applicable.	Acceptable	
Peripheral device companion app	App's core function is maintaining a persistent connection with the peripheral device for the purpose of providing the peripheral device internet access.	If applicable.	Acceptable	
	App only needs to connect to a peripheral device periodically to sync, or only needs to connect to devices, such as wireless headphones, connected via standard Bluetooth profiles.	If applicable.	Not Acceptable	

Customer 2

Healthcare task assignment system: “Final” concept

- Moving to Marshmallow: Push messaging requirements:
  - Device MQTT server remain running, requires CPU
    - **Hold a partial wake lock to ensure the CPU continues to run**
  - Device continues to have Wi-Fi access
    - **Hold a wifi lock to ensure the device does not drop WiFi**
  - Application cannot be subject to Doze mode / App Standby
    - **Application is placed on the battery whitelist**
      - Note: Customer is side-loading the application.
- Battery life is not a big concern for this customer

Customer 2

Healthcare task assignment system: “Final” concept moving forward

- Moving beyond Marshmallow???
  - Android Nougat introduced changes to Doze mode – probably OK
  - Android Oreo introduced additional restrictions
    - How do Oreo background restrictions affect the customer app?
  - Android Pie introduced additional restrictions
    - How do Pie restrictions affect the customer app?
  - Android Q…..

Customer 2

Healthcare task assignment system: “Final” concept moving forward

- How do **Oreo** background restrictions affect the customer app?
  - Oreo background restrictions place limits on running services in the background
  - Combination of whitelisting the app and holding a wake lock allowed the app to continue to run

- How do **Pie** restrictions affect the customer app?
  - “Apps that are on the Doze whitelist are exempted from the app standby bucket-based restrictions”
  - Prevent user from invoking ‘App Restrictions’

Customer 3

Transportation fleet device upgrade: Problem Statement

- Logistics organization want to update their entire fleet of corporate-owned devices
- Need latest OS to take advantage of security patches, features & user expectations
- Do not want to consider new background behaviour of the new application
  - Insufficient time to make app changes
  - Too much risk involved
  - Not prioritized work

Customer 3

Transportation fleet device upgrade: Initial concept

- Just update a few test devices and see what breaks
  - Leave the device for 1 – 2 hours to ensure Doze mode kicks in
  - Manually whitelist applications
  - Piecemeal monitoring of battery consumption
- Employees bringing their own devices (BYOD) had to follow explicit instructions to whitelist the company app

Customer 3

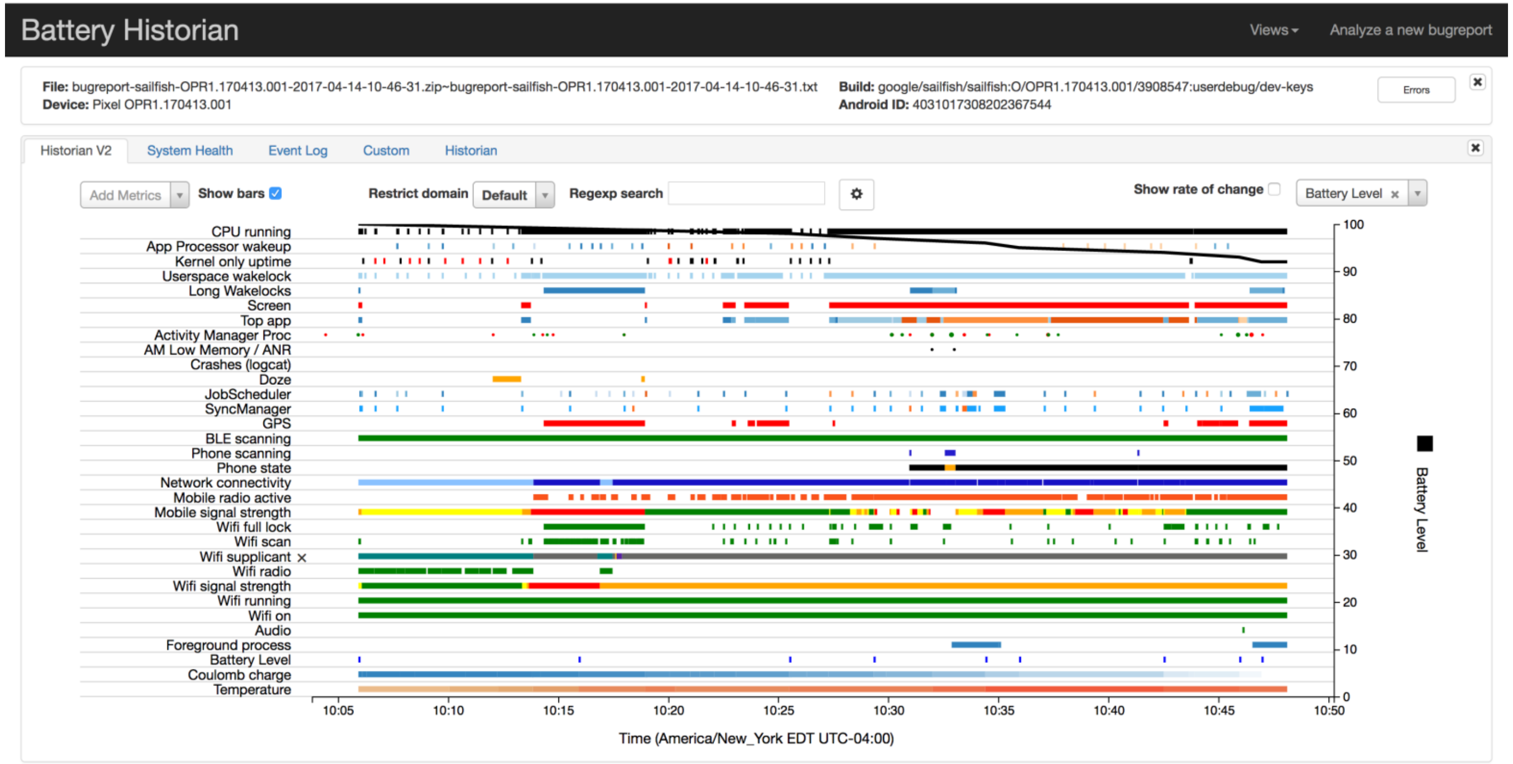
Transportation fleet device upgrade: Final concept

- We had to work with this customer & provide an OEM specific solution to ignore the effects of doze mode
  - Not recommended as a general solution!
- Customer will at some point have to repay this technical debt
  - Otherwise they are locked into the OEM specific solution
  - This solution only available to a minority of customers
  - Every new version of Android brings more risk to the customer
- Battery drain could be lessened by considering Android background app behaviour

Measuring battery consumption

Battery Historian

- I am frequently asked how to monitor device battery levels or how much extra battery use will result from e.g. taking a wake lock.
- Battery Historian is your friend:



File: bugreport-sailfish-OPR1.170413.001-2017-04-14-10-46-31.zip~bugreport-sailfish-OPR1.170413.001-2017-04-14-10-46-31.txt
Device: Pixel OPR1.170413.001

Build: google/sailfish/sailfish:O/OPR1.170413.001/3908547:userdebug/dev-keys
Android ID: 4031017308202367544

Show bars ☑ Restrict domain Default Regexp search Show rate of change □ Battery Level

Resources

“Further reading”

- All the Google advice given earlier in this deck
- I have a Test App which I use frequently to explore various concepts such as whitelist behaviour under Oreo / Pie
- My blog post aimed at Enterprise use cases and ruggedized devices, written specifically to address the concerns of enterprise customers but has implications for consumer apps too
- Reach out to me on my website or twitter

APPFORUM 2019 | ELEVATING ENTERPRISE INTELLIGENCE

Gain an intelligent edge in Enterprise IT and software development
developer.zebra.com

SAVE THE DATE

EUROPE APPFORUM 2019 | ELEVATING ENTERPRISE INTELLIGENCE

ZEBRA CAPTURE YOUR EDGE

4-6 JUNE, 2019 | WARSAW

ASIA-PACIFIC APPFORUM 2019 | ELEVATING ENTERPRISE INTELLIGENCE

ZEBRA CAPTURE YOUR EDGE

AUGUST 13-14, 2019 | SYDNEY

ASIA-PACIFIC APPFORUM 2019 | ELEVATING ENTERPRISE INTELLIGENCE

ZEBRA CAPTURE YOUR EDGE

AUGUST 20-21, 2019 | SINGAPORE

AMERICAS APPFORUM 2019 | ELEVATING ENTERPRISE INTELLIGENCE

ZEBRA CAPTURE YOUR EDGE

OCTOBER 1-2, 2019 | LAS VEGAS

Questions

Thank You

ZEBRA