ZEBRA

Indoor Locationing, it should be easy, right?

.droidconitaly

Darryn Campbell
SW Architect, Zebra Technologies
@darryncampbell

November 28th, 2020

Indoor Locationing: It should be easy, right?

Who am I?

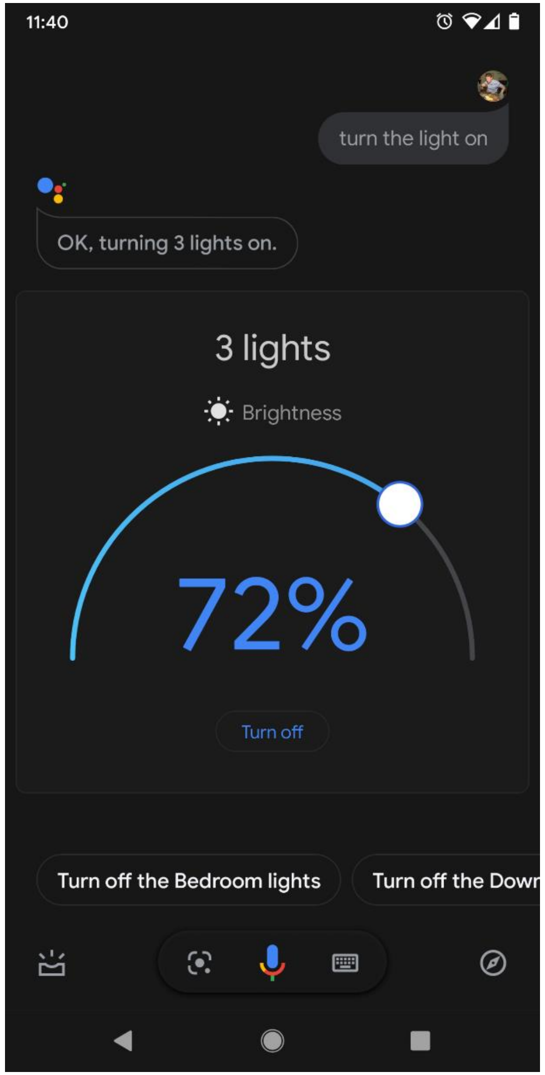
- Developer advocate / Software architect for Zebra Technologies
  - Android OEM developing task specific devices
- Responsible for our Android developer kits & APIs

@darryncampbell | darryncampbell.co.uk

Indoor Locationing: It should be easy, right?

Standard Indoor experience with Android

“OK Google, Turn the light on”



Indoor Locationing: It should be easy, right?

Available Indoor location technologies

LE

CISCO

signify

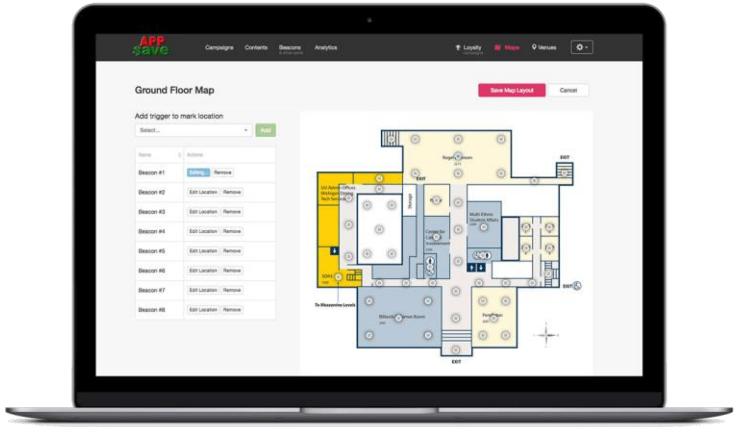
Mist
A Juniper Company

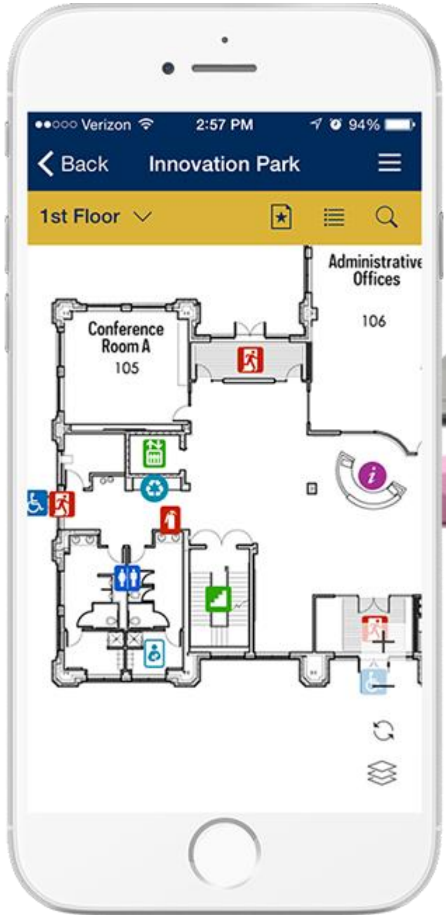
WiFi

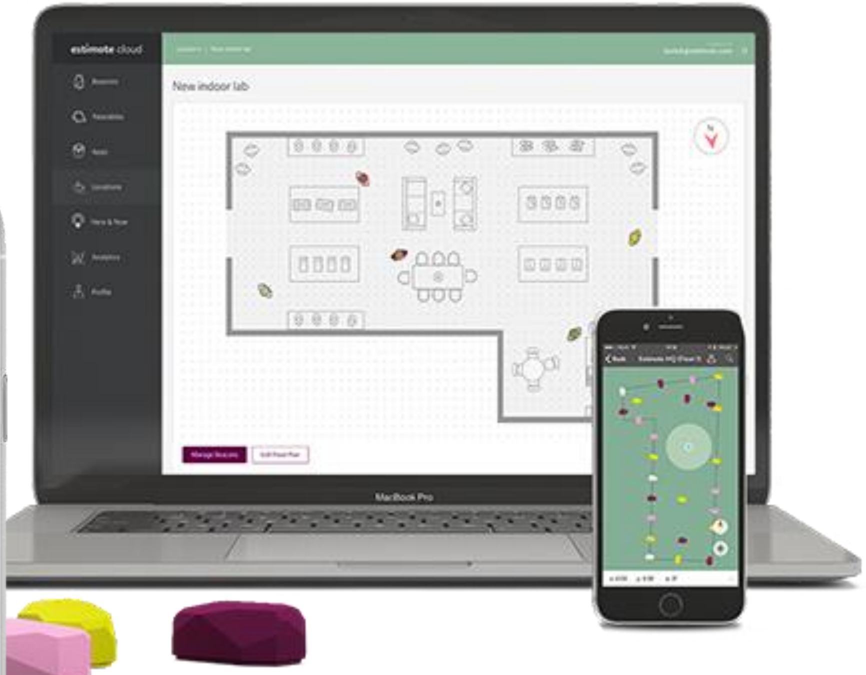
Indoor Locationing: It should be easy, right?

Backend component

All indoor location solutions include a mapping component to add context to location data points







The above are representative examples but the principle is common

Indoor Locationing: It should be easy, right?

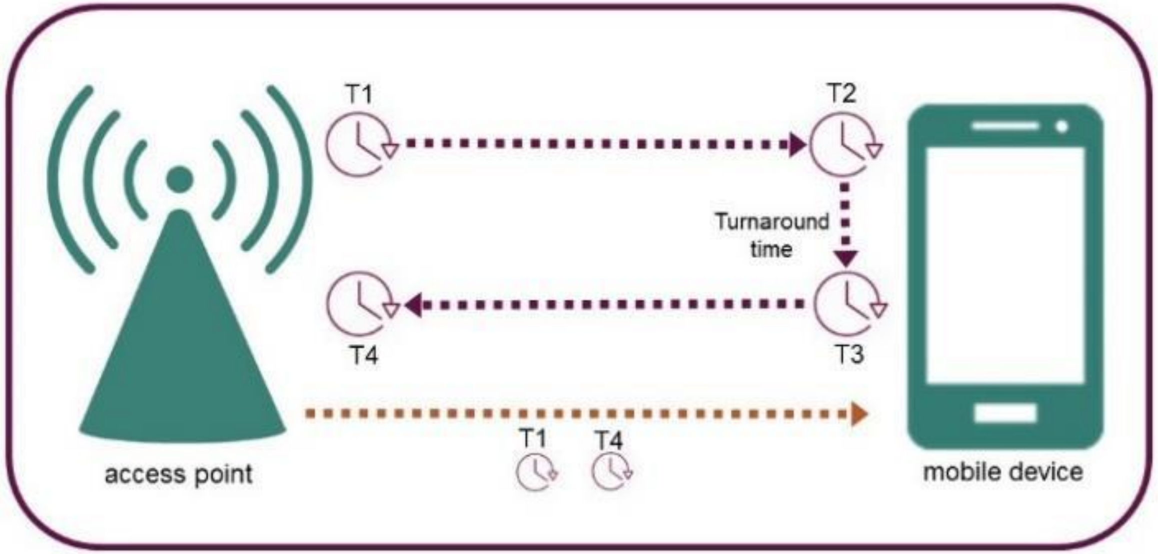
WiFi RTT: More accuracy and industry standard

- 802.11mc (IEEE standard)
- Fine Time Measurement (FTM) (WiFi Alliance)
- Round Trip Time (WiFi RTT) (Android Pie+)

How does it work?

- Device requests AP to respond with time stamps
- Round trip time is resolved to ~1ns

SPEED OF LIGHT: 1 nS = 0.3M



True round trip time of flight = T4 - T1 - T2 + T3

Indoor Locationing: It should be easy, right?

802.11mc: Required hardware / software (2019)

Support required by:

- Your access points
  - Fitlet2 (top) or WILD come up frequently in search
  - MESH routers don't officially claim support but seem to work
- Your device OS (support from Android Pie)
- Your device network stack (I tested on Pixel 2)

Indoor Locationing: It should be easy, right?

802.11mc: Required hardware / software (2020)

Most APs will use one of two SOC providers:

- Broadcom / Qualcomm

Both of these vendors have had 802.11mc support since 2018

AP manufacturers need to enable support but most have been reluctant as they already have their own proprietary solutions. Starting to change:

- Aruba recently announced they were enabling it (jointly with Zebra):
  - https://blogs.arubanetworks.com/spectrum/a-new-way-to-add-indoor-location-context/
- Industry is undergoing a hardware refresh due to WiFi6 & pressuring AP manufacturers to enable 802.11mc

Indoor Locationing: It should be easy, right?

802.11mc: Required hardware / software - 340+ Android phone models

Phones

Manufacturer and Model	Android Version
Xiaomi Mi 10 Pro	9.0+
Xiaomi Mi 10	9.0+
Xiaomi Redmi Mi 9T Pro	9.0+
Xiaomi Mi 9T	9.0+
Xiaomi Mi 9	9.0+
Xiaomi Mi Note 10	9.0+
Xiaomi Mi Note 10 Lite	9.0+
Xiaomi Redmi Note 9S	9.0+
Xiaomi Redmi Note 9 Pro	9.0+
Xiaomi Redmi Note 8T	9.0+
Xiaomi Redmi Note 8	9.0+
Xiaomi Redmi K30 Pro	9.0+
Xiaomi Redmi K20 Pro	9.0+
Xiaomi Redmi K20	9.0+
Xiaomi Redmi Note 5 Pro	9.0+

Access Points

Manufacturer and Model
Compulab WILD AP
Google Wi-Fi
Google Nest Wi-Fi Router
Google Nest Wi-Fi Point

https://developer.android.com/guide/topics/connectivity/wifi-rtt#supported-devices


SHOW ME WHAT YOU GOT!

Indoor Locationing: It should be easy, right?

802.11mc: Proof of concept

- Simulated retail environment (US)
  - Lots of shelving / items
- Google Pixel device running Pie
- 6 x Google WiFi APs
- Some assumptions wrt height from ground
- Custom application on device

Indoor Locationing: It should be easy, right?

802.11mc: Android APIs

Sample app from Google: https://github.com/android/connectivity-samples/tree/master/WifiRttScan

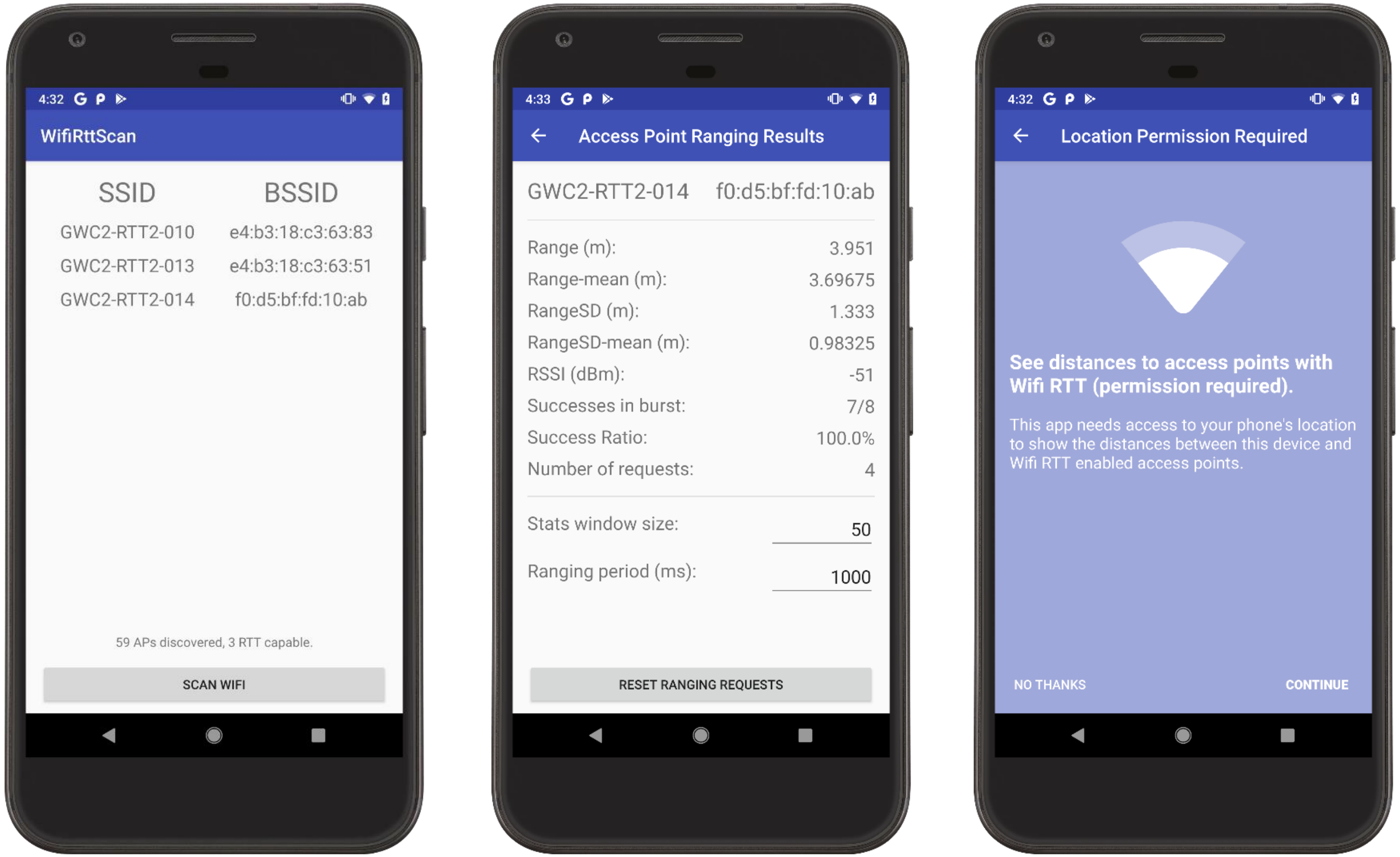
API	Purpose	Introduced (API Level)
WifiRttManager	Create objects & initialize	Pie (28)
RangingRequest RangingRequest.Builder	Request the distance to an AP	Pie (28)
RangingResult RangingResultCallback	Receive the distance to an AP	Pie (28)
ResponderLocation CivicLocationKeys	Locate the actual position of the AP	10 (29)

Indoor Locationing: It should be easy, right?

802.11mc: Official sample app

Sample app from Google: https://github.com/android/connectivity-samples/tree/master/WifiRttScan

Play Store: https://play.google.com/store/apps/details?id=com.google.android.apps.location.rtt.wifirttscan&hl=en_US

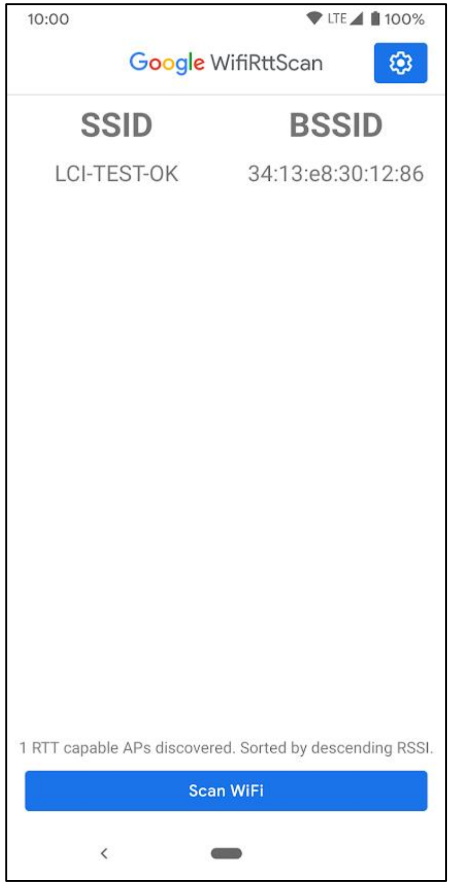


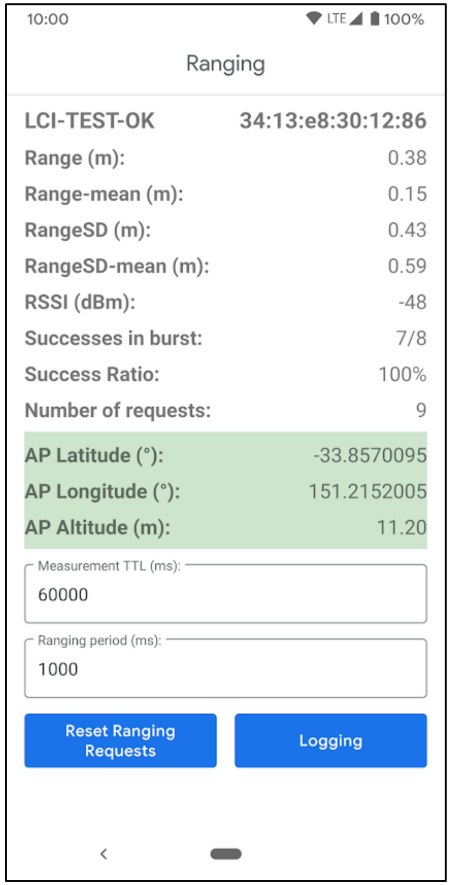
Indoor Locationing: It should be easy, right?

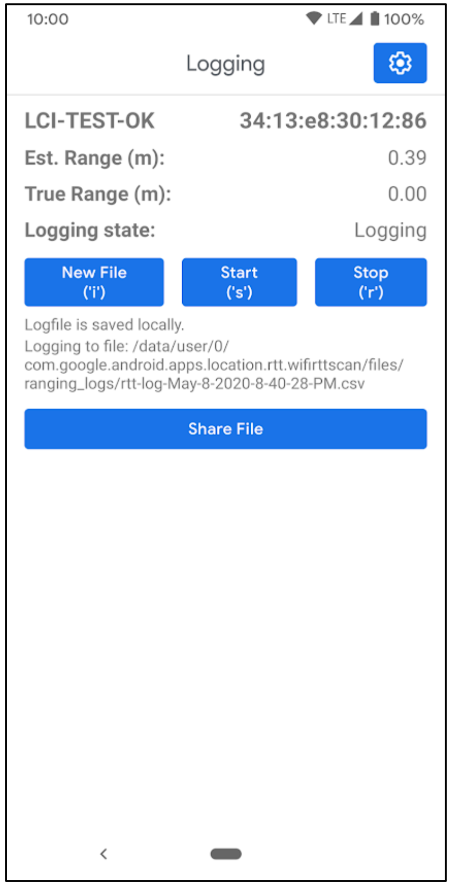
802.11mc: Official sample app

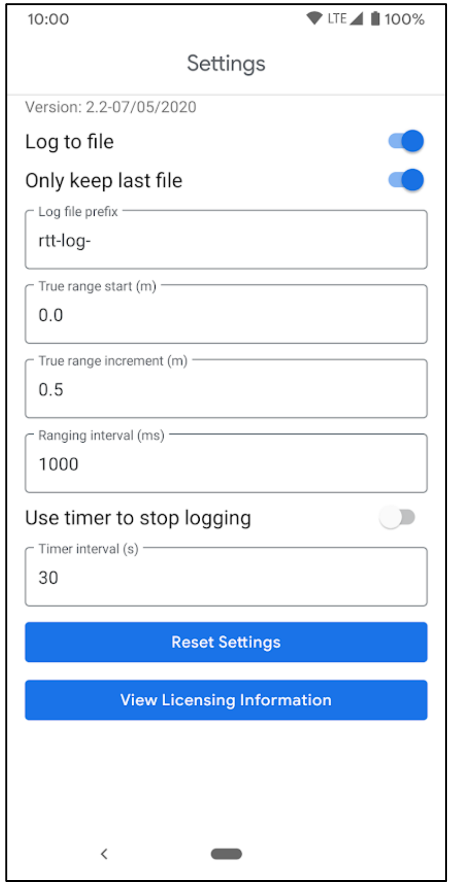
Sample app from Google: https://github.com/android/connectivity-samples/tree/master/WifiRttScan

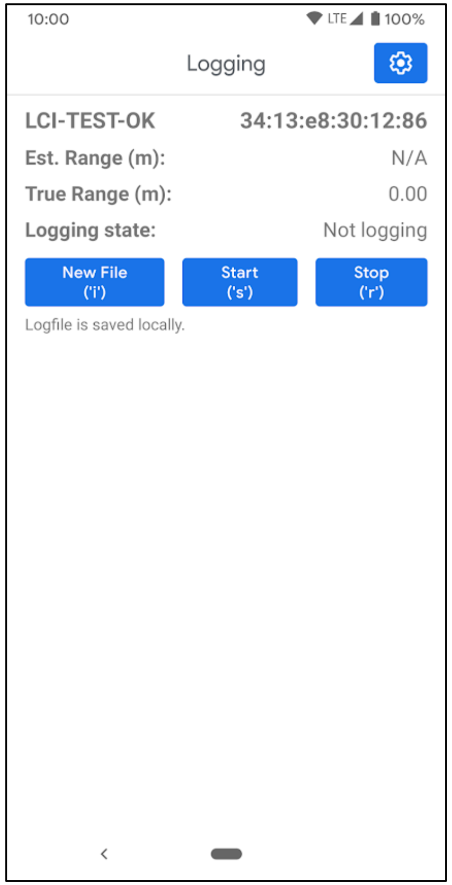
Play Store: https://play.google.com/store/apps/details?id=com.google.android.apps.location.rtt.wifirttscan&hl=en_US











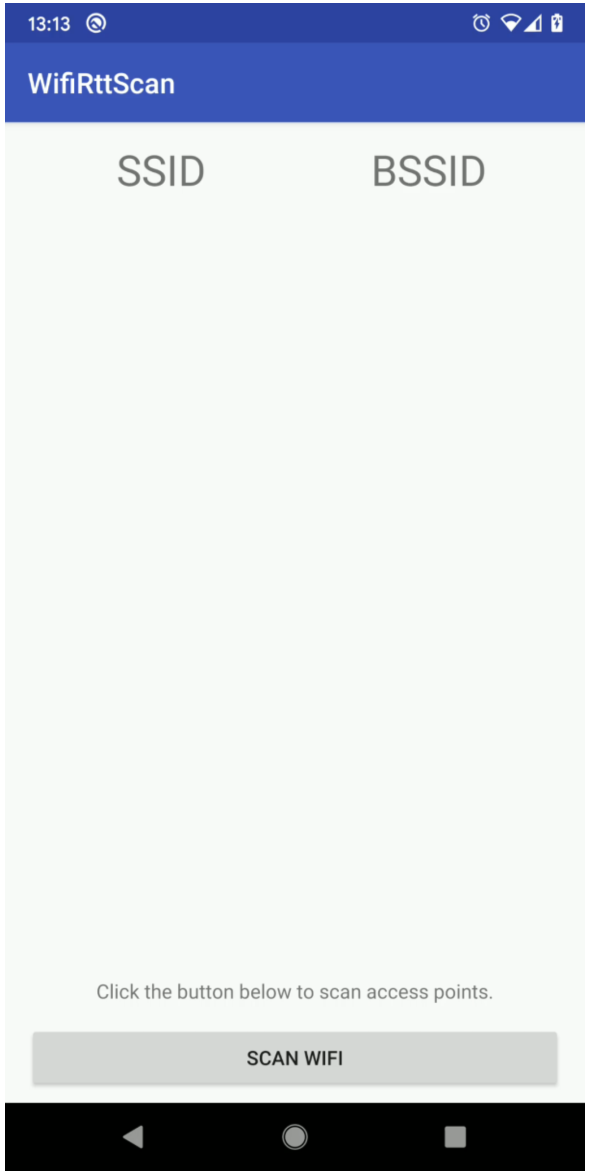
Indoor Locationing: It should be easy, right?

802.11mc: Official sample app

- Single Google WiFi AP
- Stationary Pixel 2 device (Android 10)

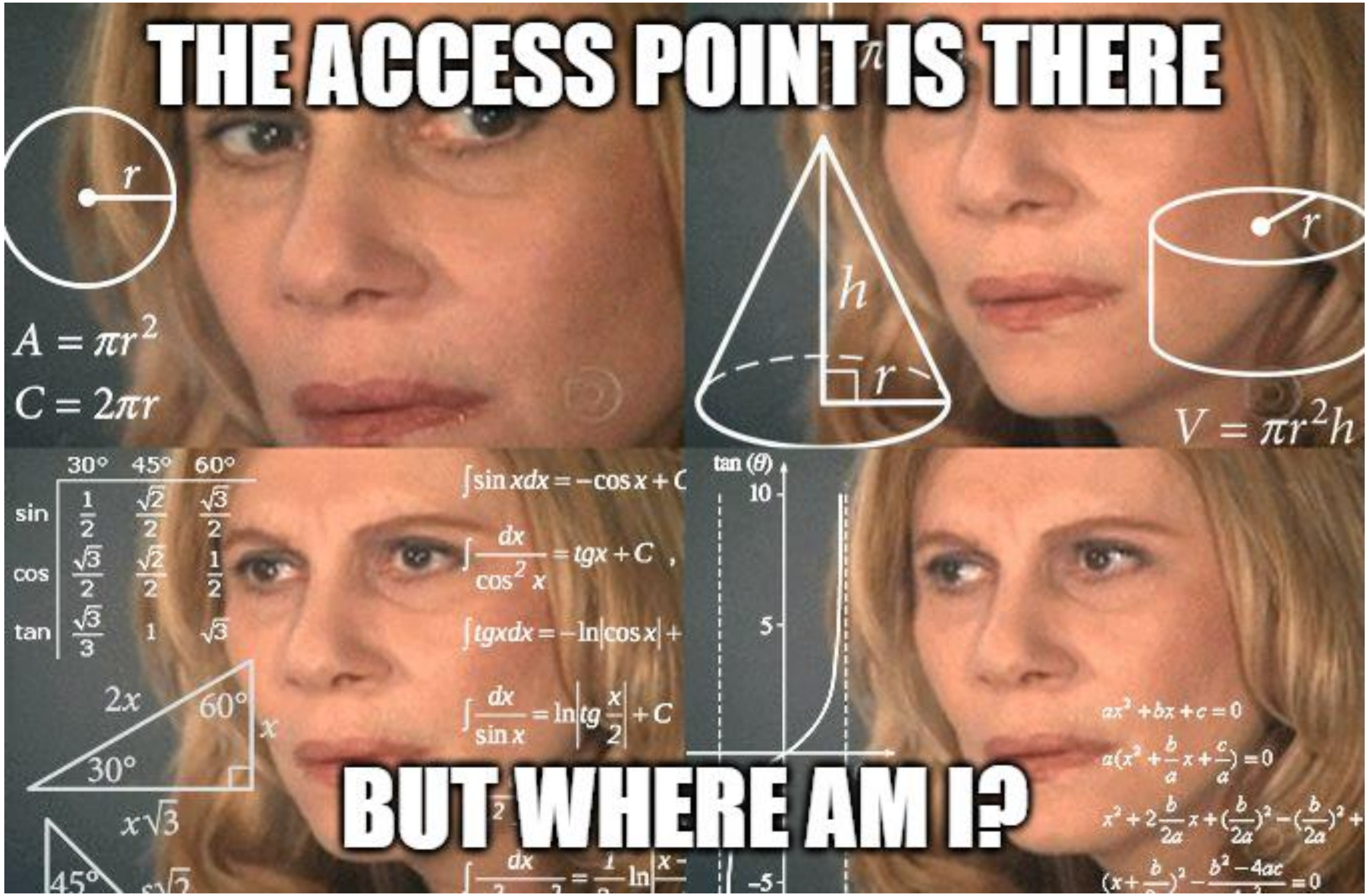
Demo is fine for:

- Showing available values returned from API
- Getting a rough distance to an AP



Indoor Locationing: It should be easy, right?

802.11mc: Turning distance into position



Indoor Locationing: It should be easy, right?

Arithmetic vs. Weighted mean: A better 'mean'ing

$$A = \frac{1}{n}\sum_{i=1}^{n} a_i = \frac{a_1 + a_2 + \cdots + a_n}{n}$$

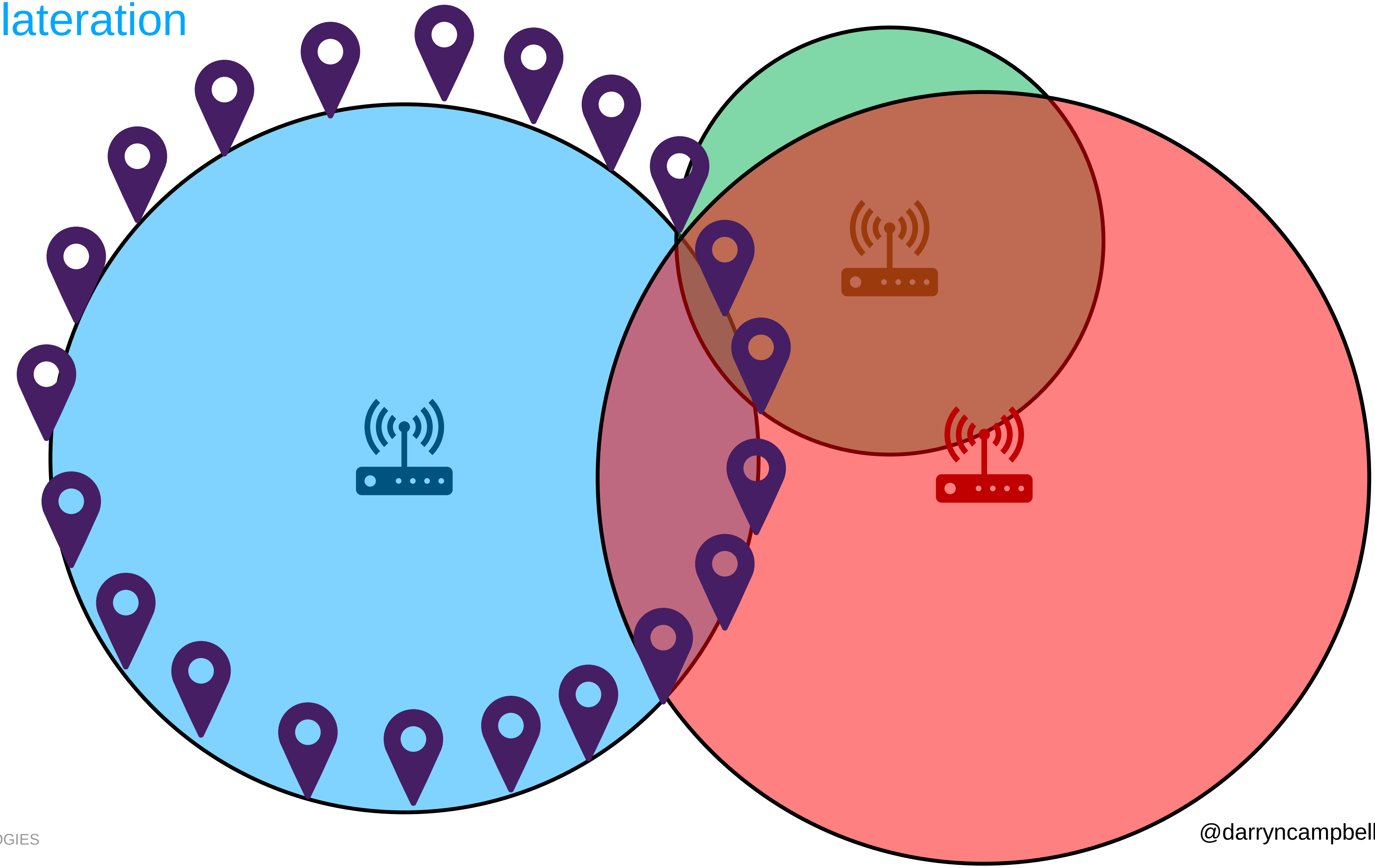
Arithmetic mean

$$\bar{x} = \frac{\sum_{i=1}^{n} \left(x_i \sigma_i^{-2} \right)}{\sum_{i=1}^{n} \sigma_i^{-2}}$$

Weighted mean
(by standard deviation)

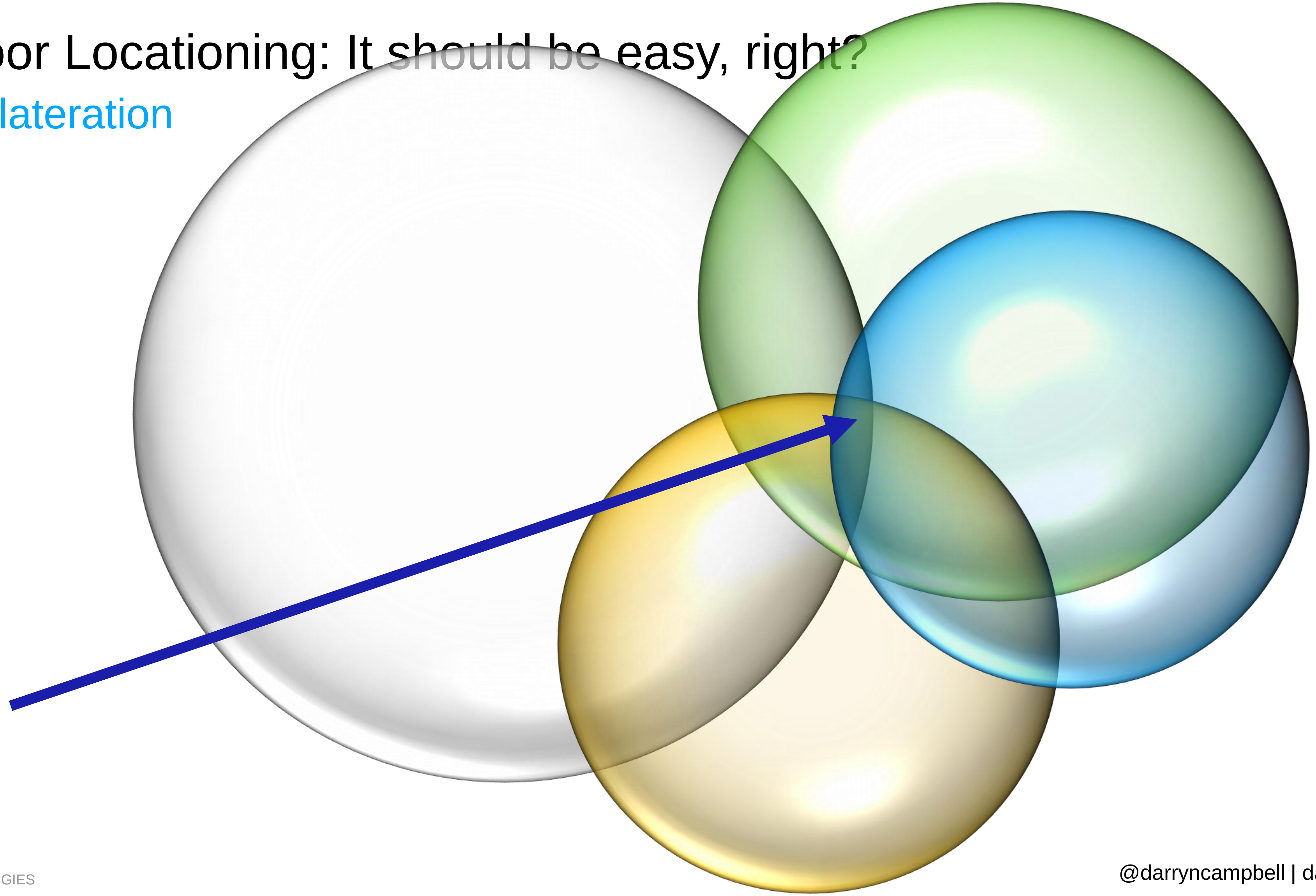
Indoor Locationing: It should be easy, right?

Multilateration



Indoor Locationing: It should be easy, right?

Multilateration



Indoor Locationing: It should be easy, right?

Multilateration

The real world is never perfect – *several multilateraion libraries exist*

$$0 = xA_m + yB_m + zC_m + D_m$$

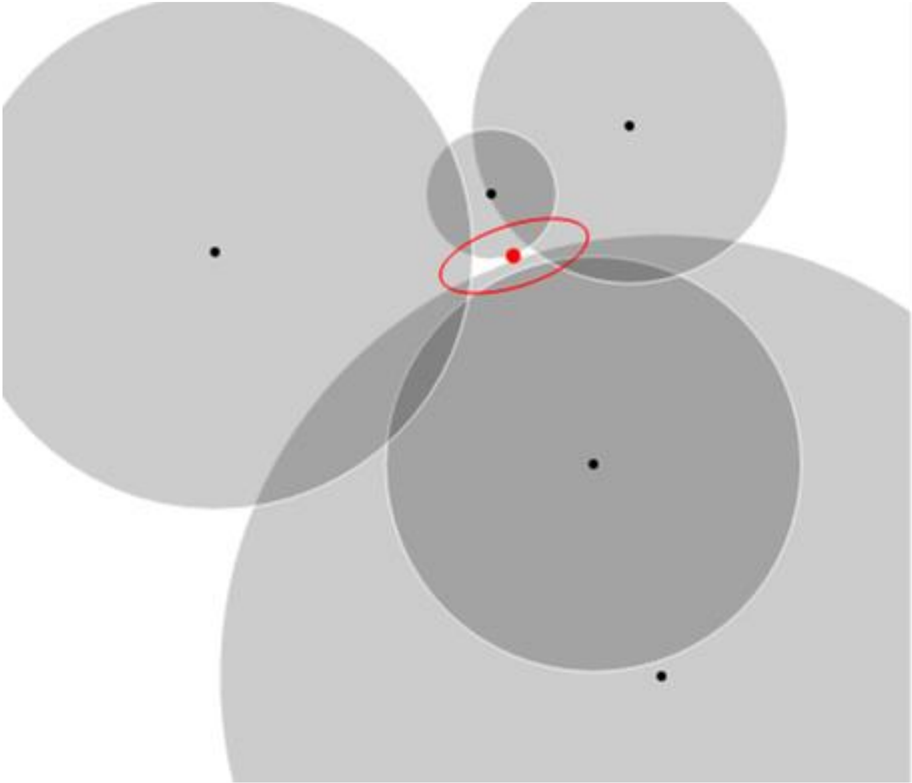
$$A_m = \frac{2x_m}{v\tau_m} - \frac{2x_1}{v\tau_1}$$

$$B_m = \frac{2y_m}{v\tau_m} - \frac{2y_1}{v\tau_1}$$

$$C_m = \frac{2z_m}{v\tau_m} - \frac{2z_1}{v\tau_1}$$

$$D_m = v\tau_m - v\tau_1 - \frac{x_m^2 + y_m^2 + z_m^2}{v\tau_m} + \frac{x_1^2 + y_1^2 + z_1^2}{v\tau_1}.$$

Example of some of the mathematics involved - from Wikipedia :/

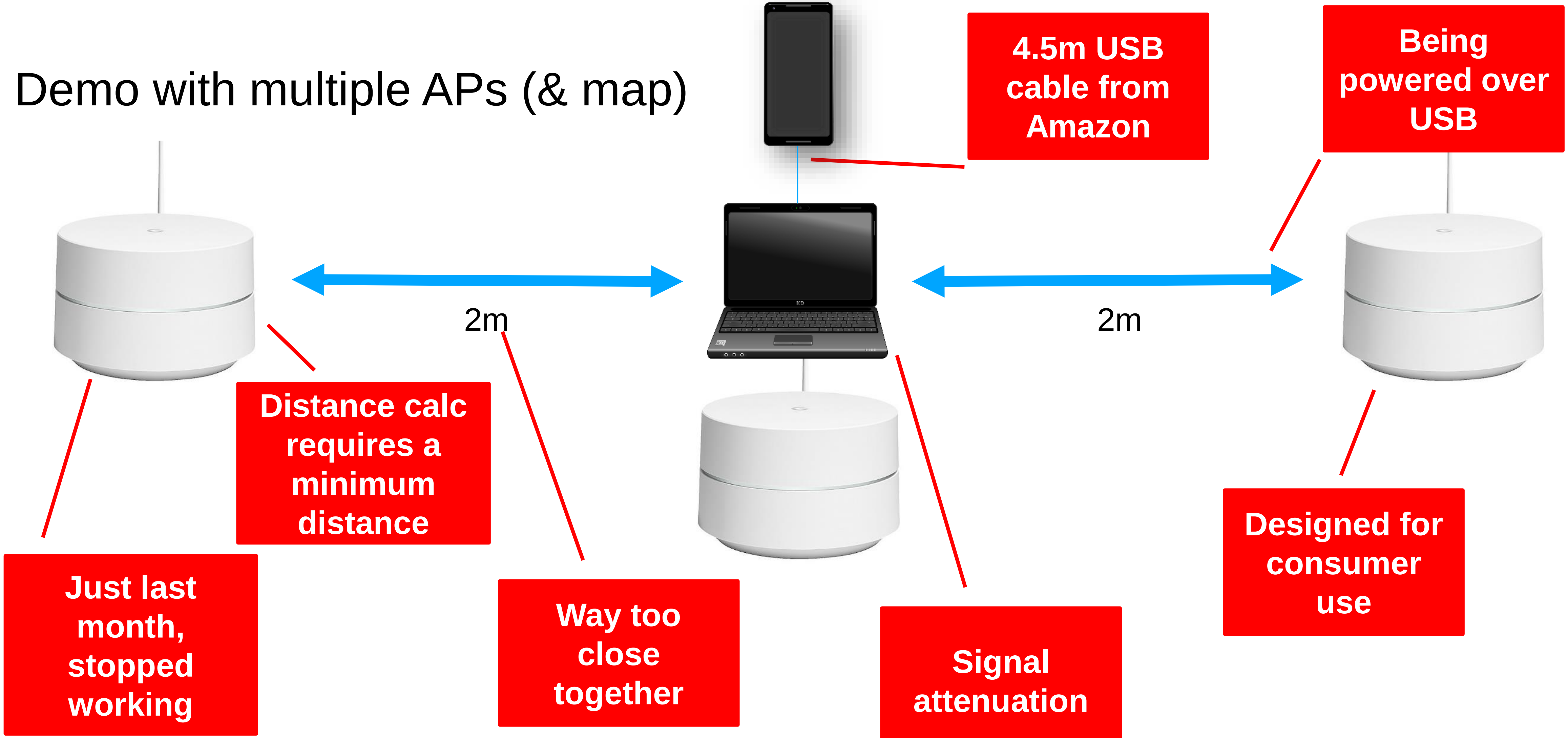


Credit: https://github.com/lemmingapex/trilateration

Indoor Locationing: It should be easy, right?

Live Demo!!

Demo with multiple APs (& map)

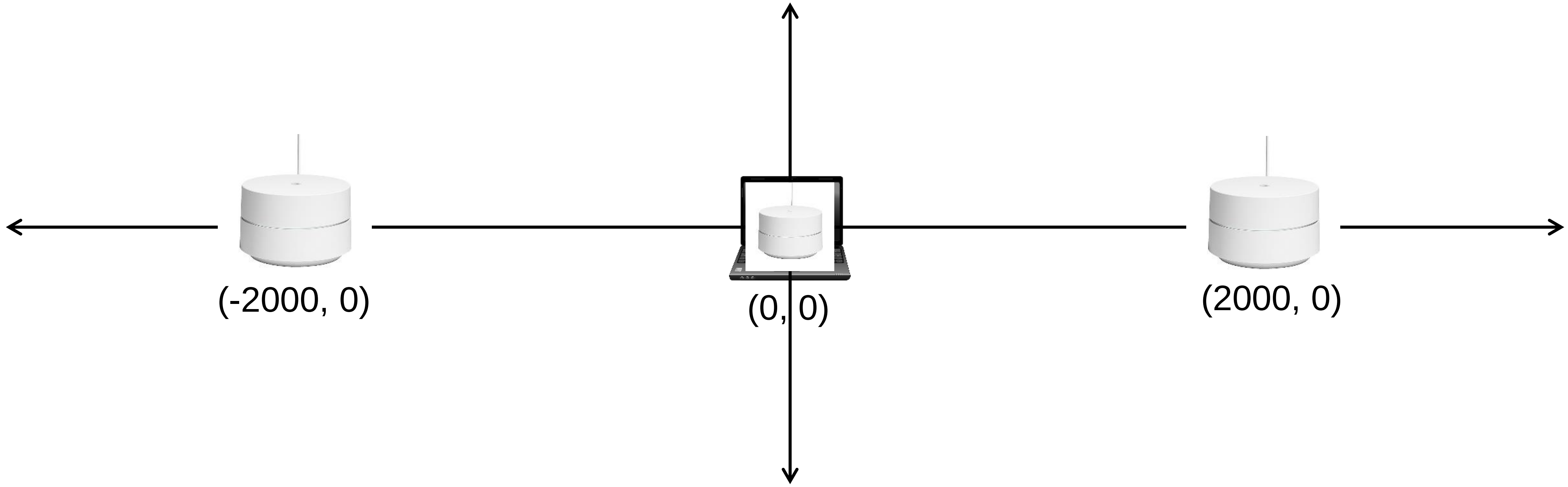


Indoor Locationing: It should be easy, right?

Live Demo!!

Define coordinate system

- 1 unit is 1mm. Origin is my laptop
- *Could use World Geodetic System* but height can get confusing

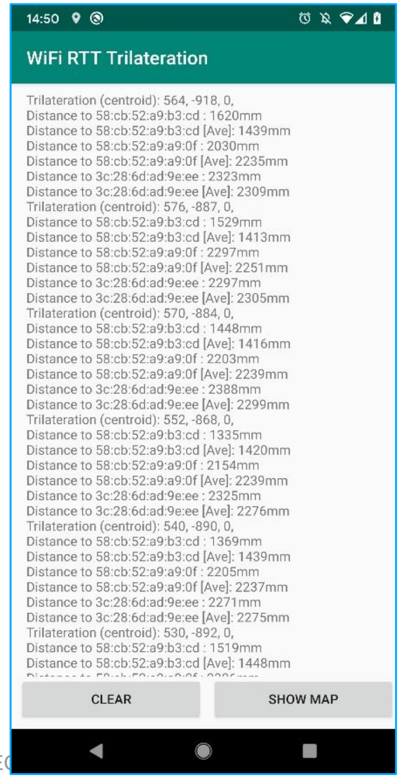


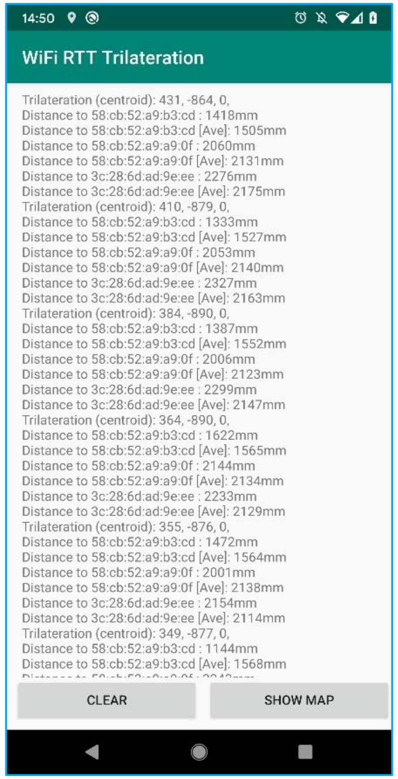
Indoor Locationing: It should be easy, right?

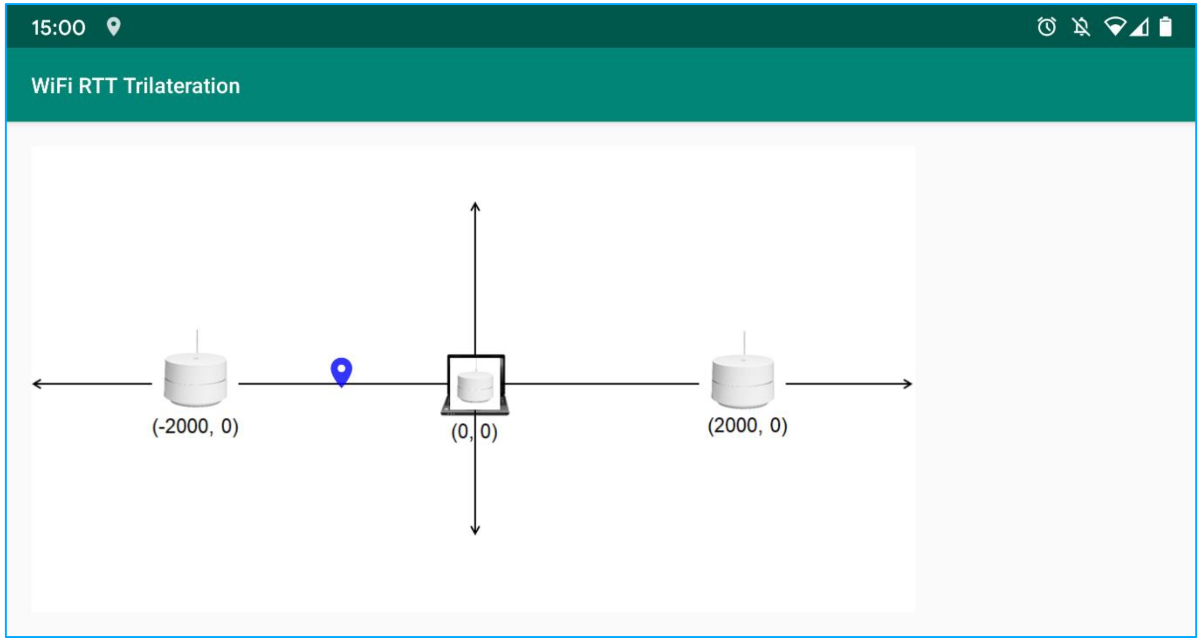
Live Demo!!

The application: https://github.com/darryncampbell/WiFi-RTT-Trilateration

- Uses code from https://github.com/lemmingapex/trilateration and [Google's sample](#)
- Very simple map – distances are converted to our own coordinate system and displayed



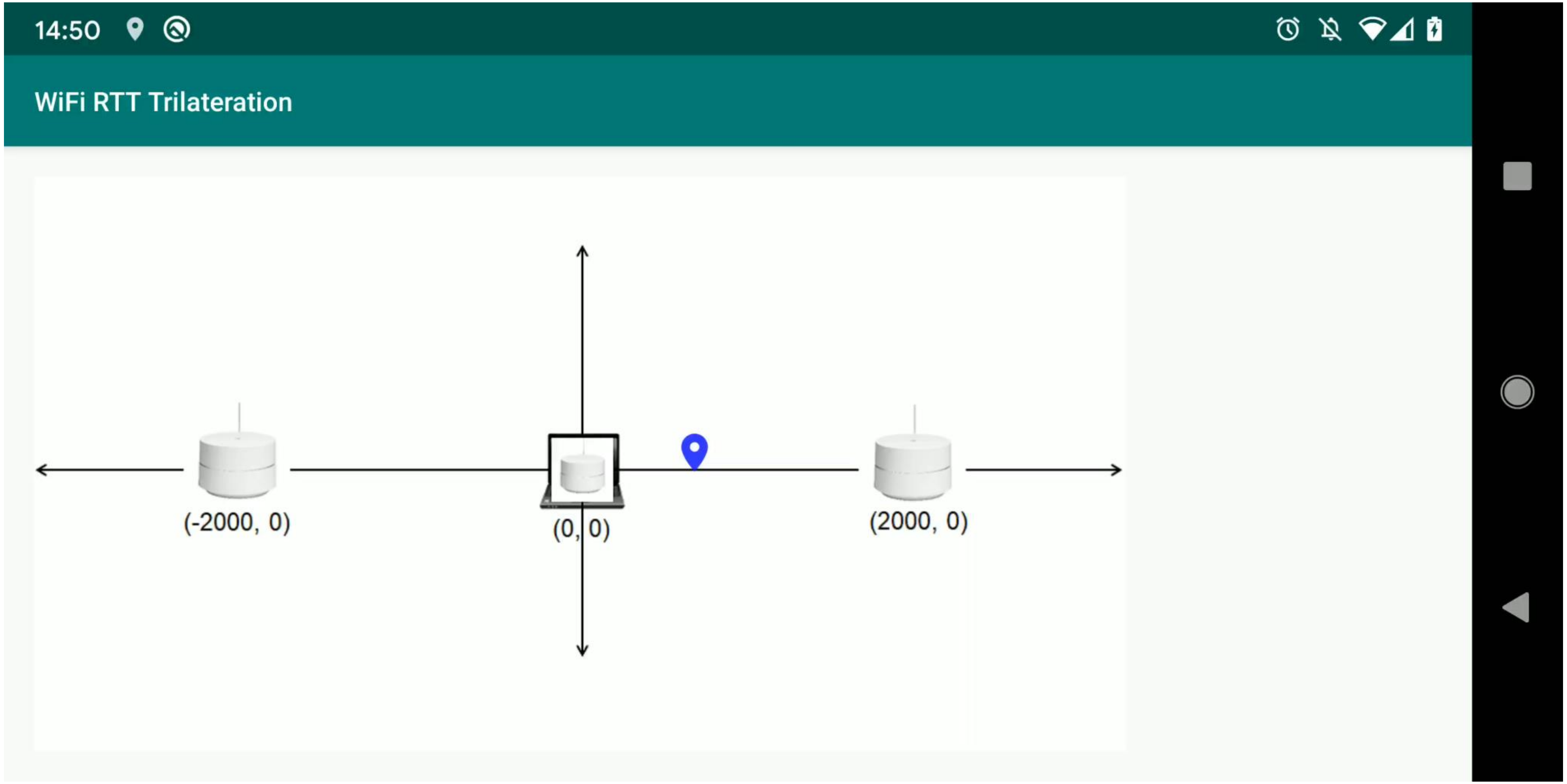




Indoor Locationing: It should be easy, right?

Live Demo!!

Video



Indoor Locationing: It should be easy, right?

Changes made in Android 10

From the Android docs:

Implementation differences based on Android version

Wi-Fi RTT was introduced in Android 9 (API level 28). When using this protocol to determine a device's position using multilateration with devices running Android 9, you need to have access to pre-determined access point (AP) locations data in your app. It is up to you to decide how to store and retrieve this data.

On devices running Android 10 (API level 29) and higher, AP location data can be represented as ResponderLocation objects, which include latitude, longitude, and altitude. For Wi-Fi RTT APs that support Location Configuration Information/Location Civic Report (LCI/LCR data), the protocol will return a ResponderLocation object during the ranging process.

This feature allows apps to query APs to ask them for their position directly rather than needing to store this information ahead of time. So, your app can find APs and determine their positions even if the APs were not known before, such as when a user enters a new building.

Indoor Locationing: It should be easy, right?

Changes made in Android 10

- Access points need to be provisioned with location information
- No need to define your own coordinate system – use lat / long
- Still need to trilaterate your own position based on the distance from lat / long coordinates
  - Based on WGS 84
- Can provide other information such as altitude (floors / meters), height above floor
  - And even civic data (LCR) such as City, Country, Building, Desk, Post code etc.
  - API just trusts what the API reports, depends on correct provisioning.

Indoor Locationing: It should be easy, right?

Conclusions

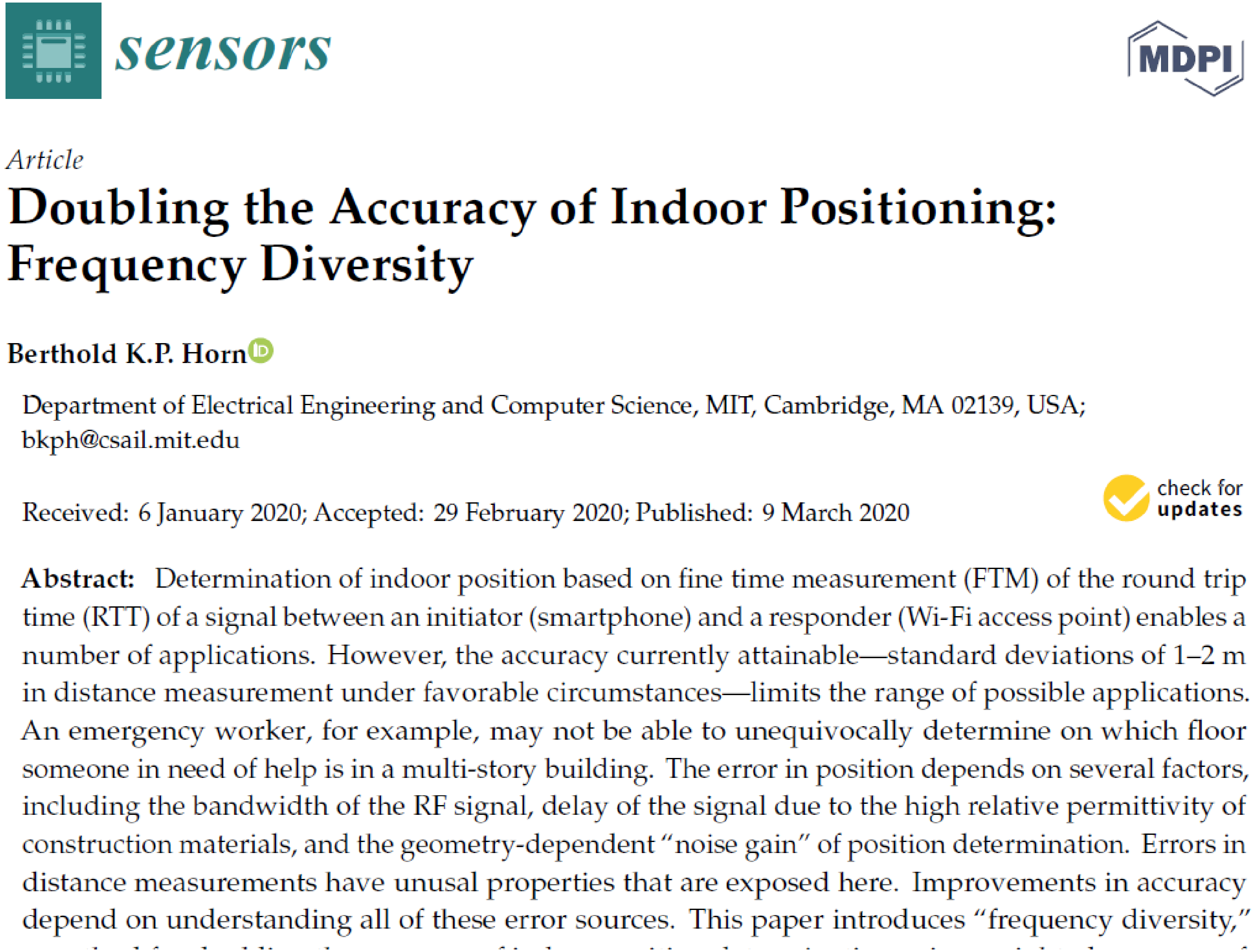
- 802.11mc is interesting but provides nowhere near the 30cm accuracy stated as the ‘theoretical maximum’
- The Android APIs easy to use but only get you part-way to finding your location, compare that with the fused location provider where your position is returned in a single call.
- Any solution needs support from both Hardware and Software making a generic deployment impractical at the present time.

Indoor Locationing: It should be easy, right?

Conclusions

Don’t just take my word for it:

https://www.mdpi.com/1424-8220/20/5/1489



sensors

MDPI

Article

Doubling the Accuracy of Indoor Positioning: Frequency Diversity

Berthold K.P. Horn

Department of Electrical Engineering and Computer Science, MIT, Cambridge, MA 02139, USA; bkph@csail.mit.edu

Received: 6 January 2020; Accepted: 29 February 2020; Published: 9 March 2020

check for updates

Abstract: Determination of indoor position based on fine time measurement (FTM) of the round trip time (RTT) of a signal between an initiator (smartphone) and a responder (Wi-Fi access point) enables a number of applications. However, the accuracy currently attainable—standard deviations of 1–2 m in distance measurement under favorable circumstances—limits the range of possible applications. An emergency worker, for example, may not be able to unequivocally determine on which floor someone in need of help is in a multi-story building. The error in position depends on several factors, including the bandwidth of the RF signal, delay of the signal due to the high relative permittivity of construction materials, and the geometry-dependent “noise gain” of position determination. Errors in distance measurements have unusal properties that are exposed here. Improvements in accuracy depend on understanding all of these error sources. This paper introduces “frequency diversity,”

Indoor Locationing: It should be easy, right?

Conclusions

Don't just take my word for it:

https://www.mdpi.com/1424-8220/20/5/1489

Abstract: Determination of indoor position based on fine time measurement (FTM) of the round trip time (RTT) of a signal between an initiator (smartphone) and a responder (Wi-Fi access point) enables a number of applications. However, **the accuracy currently attainable—standard deviations of 1–2 m in distance measurement under favorable circumstances—limits the range of possible applications**. An emergency worker, for example, may not be able to unequivocally determine on which floor someone in need of help is in a multi-story building. The error in position depends on several factors, including the bandwidth of the RF signal, delay of the signal due to the high relative permittivity of construction materials, and the geometry-dependent "noise gain" of position determination. Errors in distance measurements have unusal properties that are exposed here. **Improvements in accuracy depend on understanding all of these error sources**.

Indoor Locationing: It should be easy, right?

Resources

- Medium blog: https://medium.com/@darryncampbell_83863/indoor-positioning-with-wifi-rtt-and-google-wifi-a638f1147b84
- Android code:
  - https://github.com/android/connectivity-samples/tree/master/WifiRttScan
  - https://github.com/darryncampbell/WiFi-RTT-Trilateration
  - https://github.com/lemmingapex/trilateration
- Android docs:
  - https://developer.android.com/guide/topics/connectivity/wifi-rtt
  - https://developer.android.com/reference/android/net/wifi/rtt/package-summary.html
  - https://developer.android.com/reference/android/net/wifi/rtt/ResponderLocation
- Academic paper: https://www.mdpi.com/1424-8220/20/5/1489
- Videos shown in this presentation:
  - Wifi RTT Scan: https://youtu.be/7PQ_fL2Cy4k
  - Wifi RTT Trilateration: https://youtu.be/gTvlMw_IqPQ

Questions?

http://developer.zebra.com

Thank You

.droidconitaly

ZEBRA

Indoor Locationing: It should be easy, right?

802.11mc: Proof of concept

- Indoor office environment (US)
  - Lots of desks / chairs / cubicles
- Google Pixel device running Pie