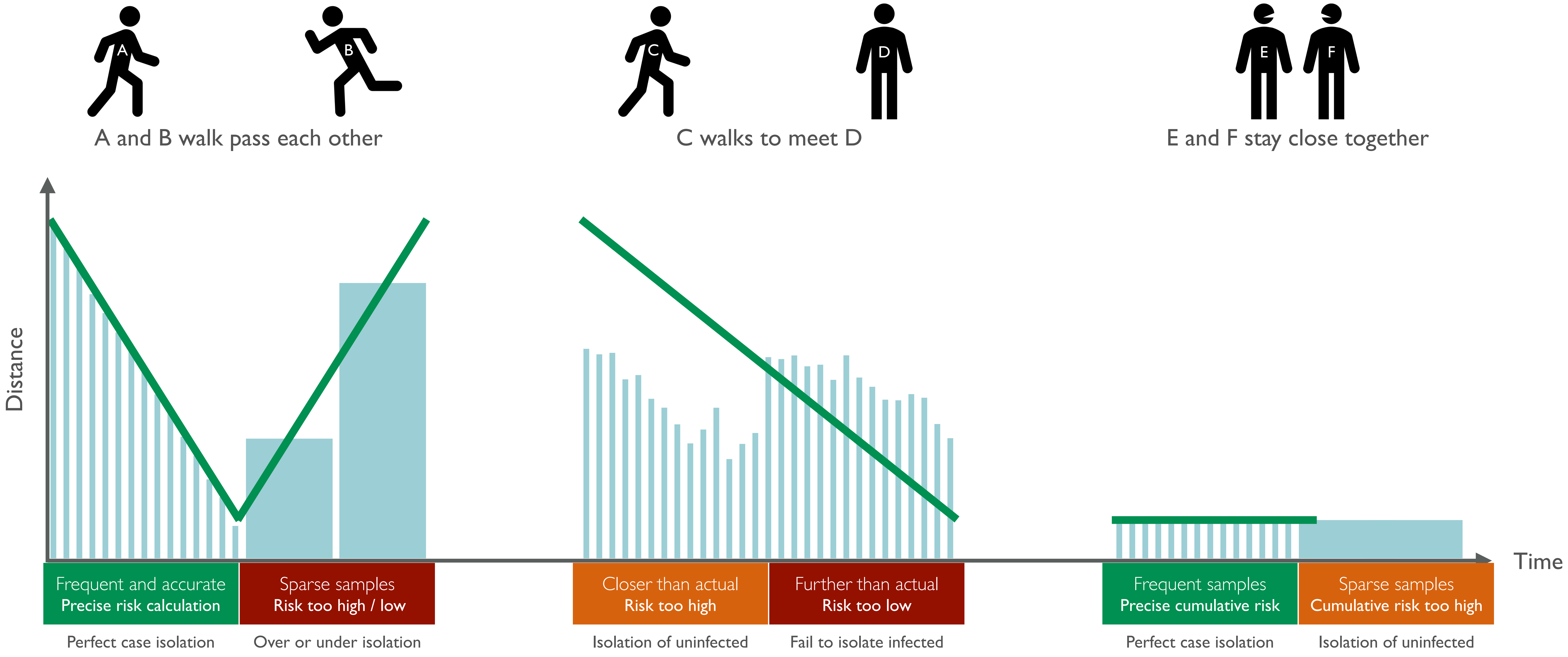
PROXIMITY DETECTION

Solution design



A and B walk pass each other

C walks to meet D

E and F stay close together

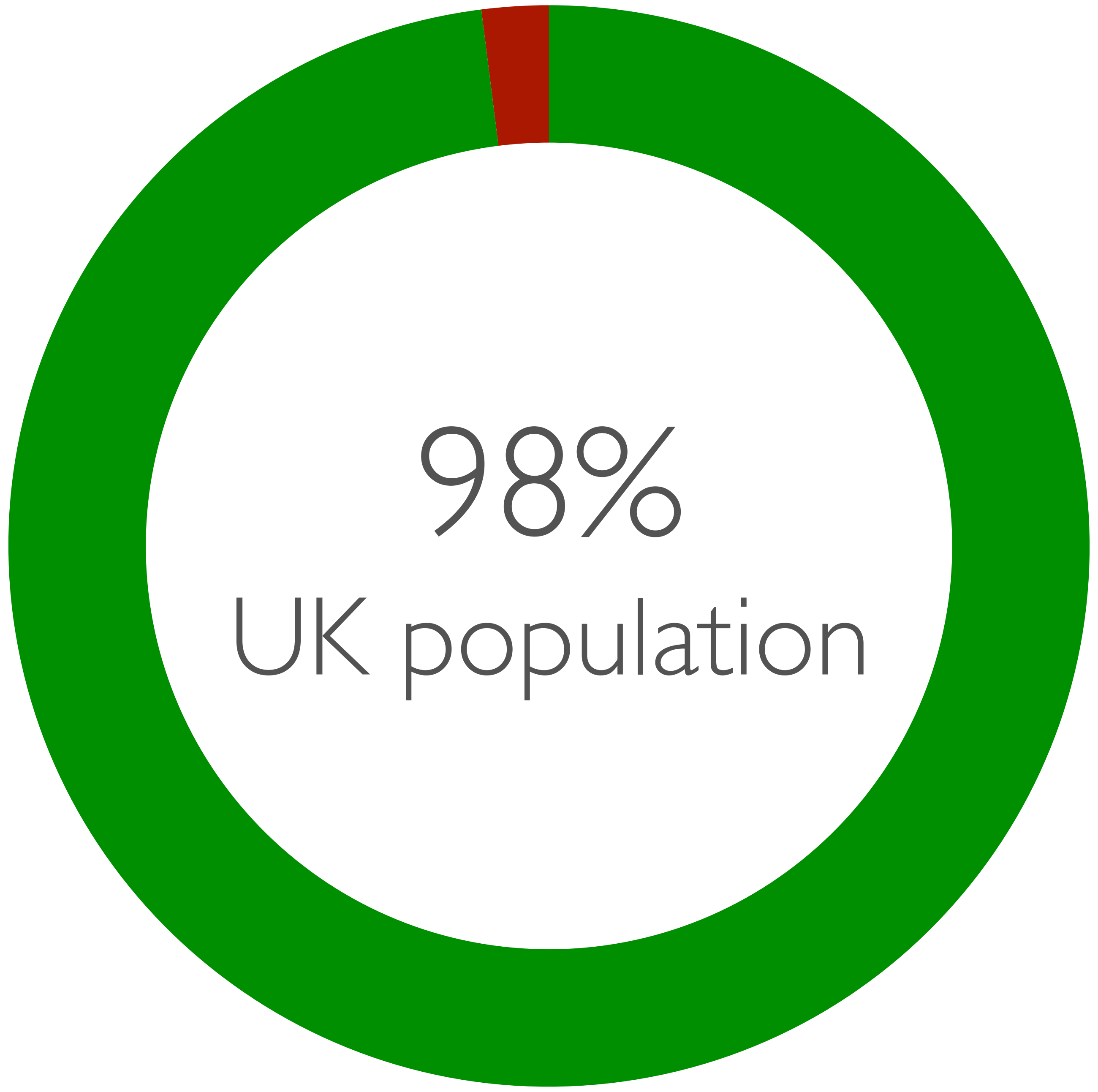
Frequent and accurate Precise risk calculation	Sparse samples Risk too high / low	Closer than actual Risk too high	Further than actual Risk too low	Frequent samples Precise cumulative risk	Sparse samples Cumulative risk too high
Perfect case isolation	Over or under isolation	Isolation of uninfected	Fail to isolate infected	Perfect case isolation	Isolation of uninfected

INFECTION CONTROL

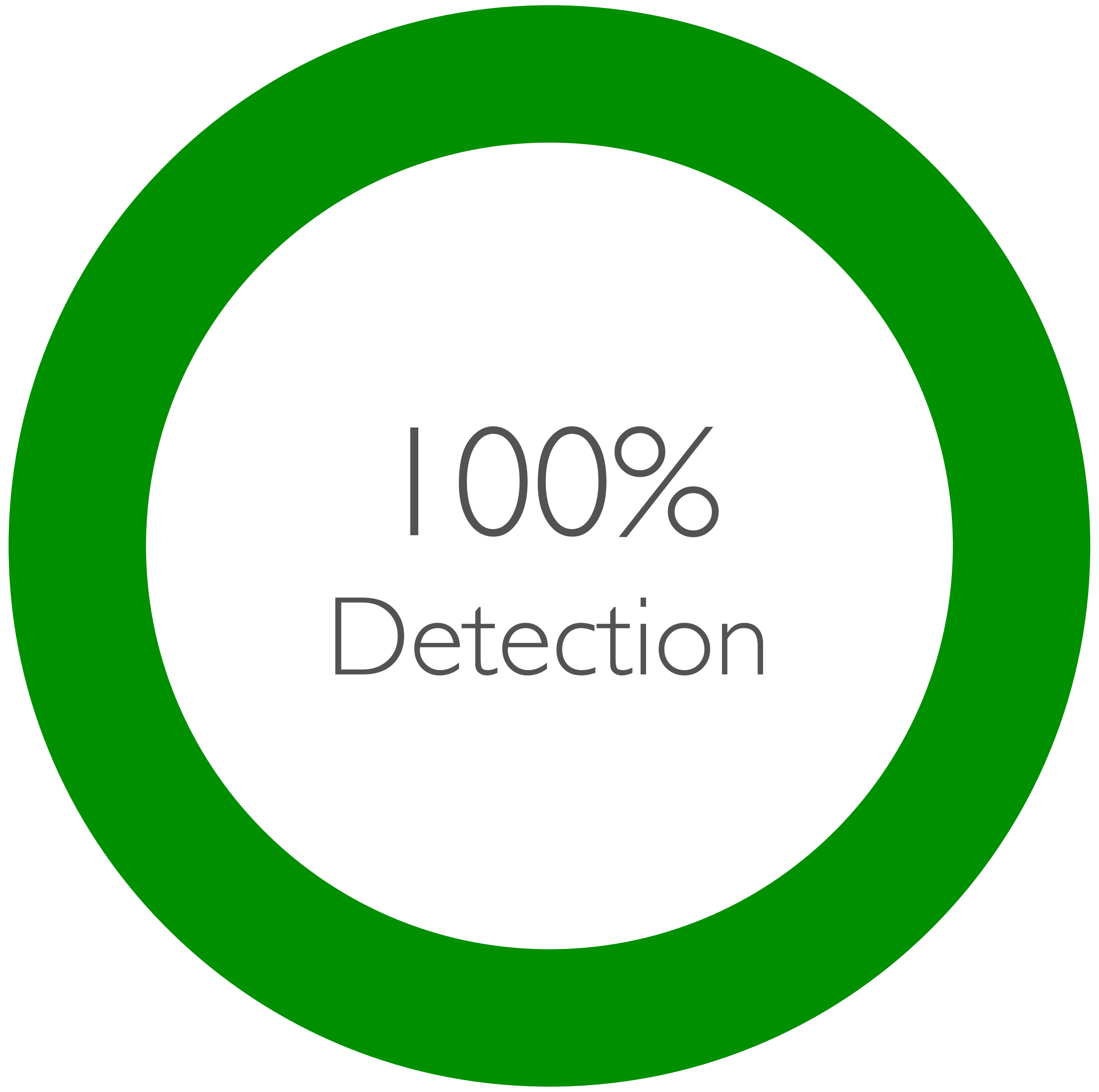
Epidemiological model of case isolation is based on cumulative exposure

INFECTION CONTROL

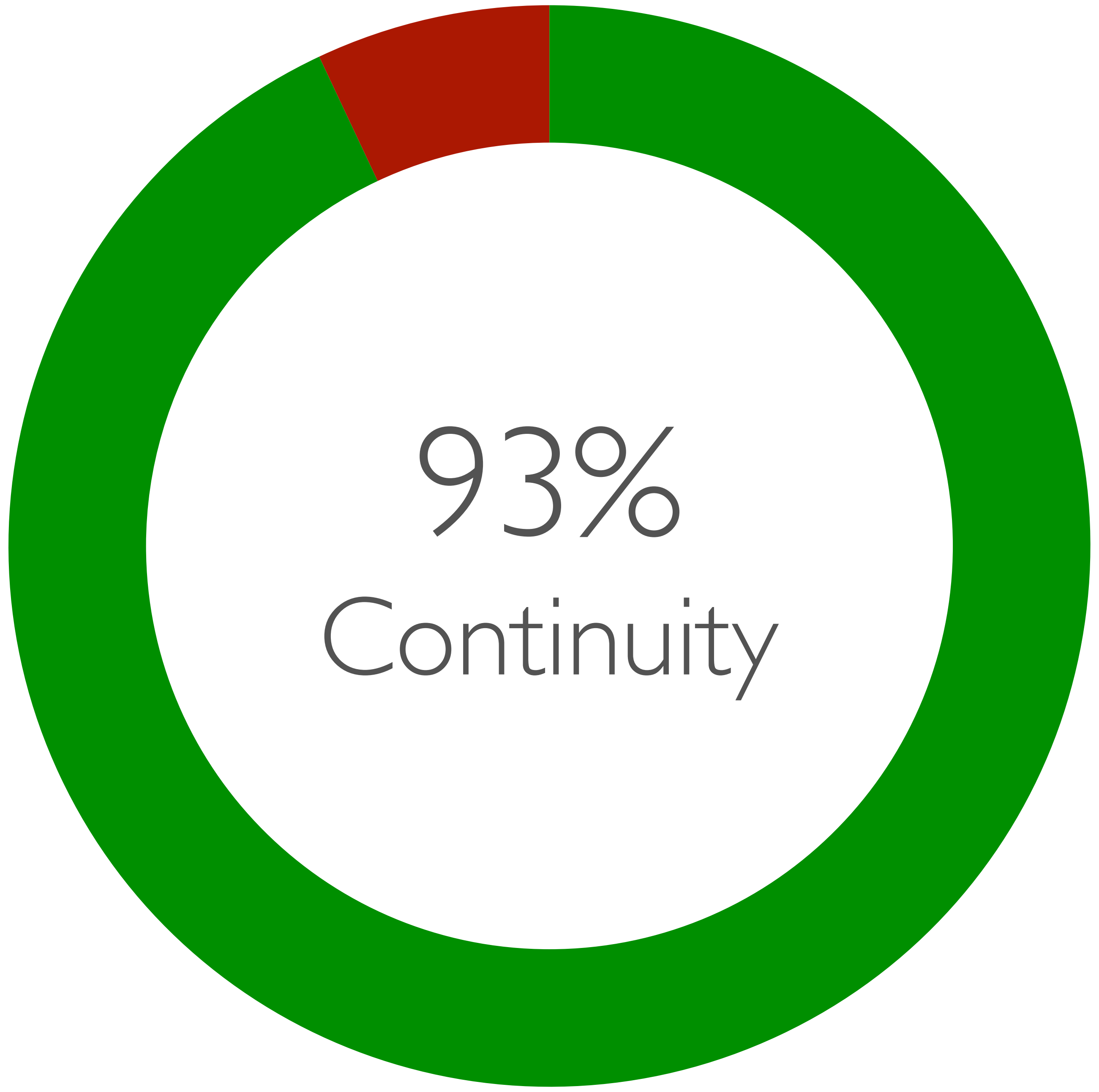
- Epidemiology model
  - Infection risk estimated based on cumulative exposure to the disease.
  - Duration and distance are key measurements for risk estimation.
  - Disease controlled by rapid isolation of infected individuals across the entire population.
- Solution requirements
  - Precise distance measurement between people within epidemiology relevant range.
  - Frequent sampling to measure time spent in close proximity for cumulative risk estimation.
  - Measurements taken for all encounters across the entire population.
- Technical requirements
  - Detection of all devices within 8 metres.
  - Distance measurement accuracy at ± 50 centimetres (or less) in 0 - 2 metre range.
  - At least one measurement per 30 seconds to quantify exposure across different activities.



What proportion of the population can use this solution?



How many devices in the vicinity are detected for contact tracing?



How many 30s windows have distance measurements for risk estimation?

DESIGN GOAL

Continuous distance measurement between all iOS and Android devices

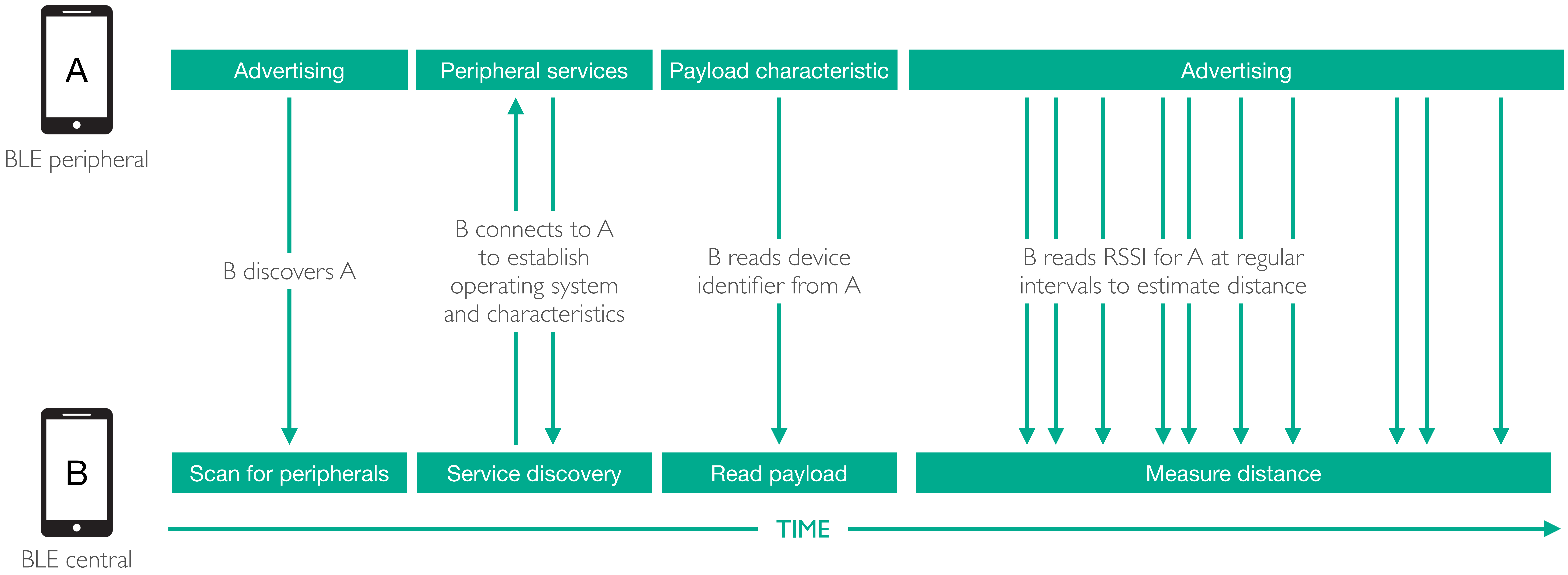
Continuity is 93% for 10 devices, 96% for 9 devices, 99% for 3 devices

DESIGN GOAL

- Epidemiology requirements
  - Population coverage for infection control across the entire nation.
  - Device detection for rapid and comprehensive contact tracing.
  - Frequent and accurate distance measurements for infection risk estimation and case isolation.

- Impact of requirements
  - Low population coverage means disease cannot be contained leading to wide area lock downs.
  - Failing to detect a device means infected person is not identified leading to more infections.
  - Sparse or inaccurate distance measurements will falsely isolate healthy people and miss infected people.

- Design targets
  - 95% of phones in the UK and worldwide are able to use the solution.
  - 99% of phones within 8 metres are detected by the solution.
  - 95% of 30 second time windows contain a distance measurement for each phone within 8 metres.



SOLUTION

Bluetooth low energy (BLE) peripheral and central

SOLUTION

- Device discovery for contact tracing
  - Discovery of Android peripherals by Android and iOS centrals is relatively trivial.
  - Discovery of iOS peripherals by iOS centrals is trivial … when the app is in foreground and device is unlocked.
  - Discovery of iOS peripherals by iOS centrals is difficult when the app is in background - **We have two solutions.**
- Payload reading for contact tracing
  - User identification data encoded in payload data … **solution is payload agnostic**.
  - Insert your own payload to implement your own security and privacy solution.
  - Solution supports centralised and decentralised approaches … **we have both example solutions.**
- Distance measurement for infection risk estimation and case isolation
  - Received signal strength indicator (RSSI) as basis for distance estimation … **more samples improves accuracy**.
  - Continuous measurement of Android and iOS peripherals by Android centrals is relatively trivial.
  - Continuous measurement of iOS peripherals by iOS centrals is difficult … **we have a solution.**

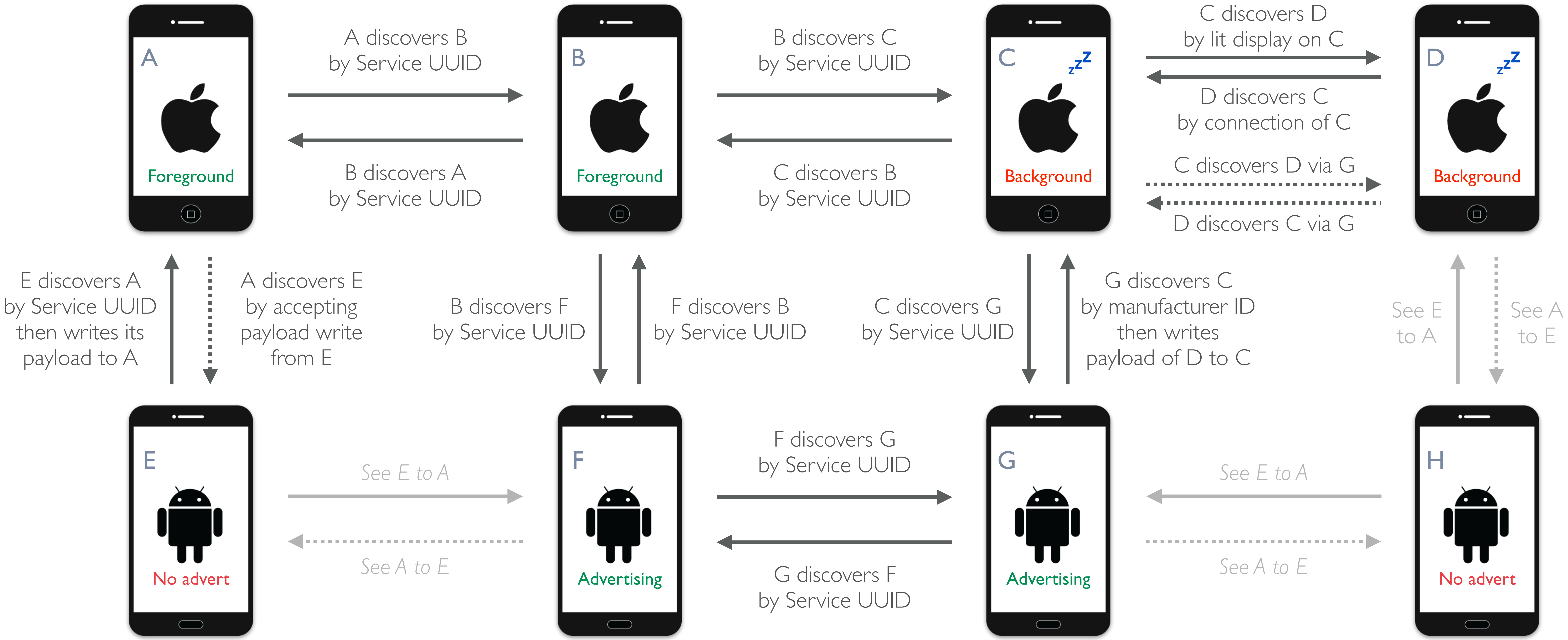
Android	Service UUID to enable discovery by Android and iOS (foreground, background) scan for peripherals Service UUID - 16 bytes	Custom ID for all Android devices as search key for pseudo device address data Manufacturer ID - 2 bytes	Random and rotating pseudo device address as alternative to BLE device address that can change at any rate Manufacturer data - 6 bytes
iOS foreground	Service UUID to enable discovery by Android and iOS (foreground, background) scan for peripherals Service UUID - 16 bytes	Predefined ID for all Apple devices Manufacturer ID - 2 bytes	
iOS background		Predefined ID for all Apple devices Manufacturer ID - 2 bytes	

ADVERTISING

Peripheral advert data regions utilised by the solution

ADVERTISING

- iOS central background scan mandates discovery by Service UUID
    - Android and iOS peripherals advertise Service UUID … **enable discovery by Android and iOS centrals**.
    - Android central scan reports all Android and iOS foreground peripherals … **enables connection free RSSI measurements**.
    - iOS central background scan only reports first discovery of iOS peripherals … **scanning again does not report known iOS peripherals**.
    - iOS central background scan reports all Android peripherals for every scan request … **enables connection free RSSI measurements**.
- iOS peripheral background advertising hides Service UUID as proprietary data
    - Android central discovers iOS background peripheral by Apple manufacturer ID then connects to confirm Service UUID … **proprietary data can change**.
    - iOS central foreground scan can interpret iOS peripheral background advert proprietary data to enable discovery … **foreground operation is rare**.
    - iOS central background scan does not discover iOS peripheral background adverts … **we have two solutions.**
- Android peripheral address can change every few hours or seconds
    - Samsung A10 device address can change every few seconds, thus every distance measurement requires read payload … **too slow, impacts continuity**.
    - Android peripheral advert includes pseudo device address that rotates at a fixed rate (15 minutes, adjustable) as custom manufacturer data.
    - Android and iOS central scan uses pseudo address to link previously read payload to Android peripheral … **increases connection free RSSI measurements**.
- Android phones may not support advertising
    - Around 35% of Android devices in the UK do not support advertising due to software or hardware limitations (e.g. Samsung J6).
    - Android and iOS centrals cannot discover a non-advertising Android peripheral … **we have a solution.**


A
Foreground
B
Foreground
C
Background
D
Background
E
No advert
F
Advertising
G
Advertising
H
No advert
A discovers B by Service UUID
B discovers A by Service UUID
B discovers C by Service UUID
C discovers B by Service UUID
C discovers D by lit display on C
D discovers C by connection of C
C discovers D via G
D discovers C via G
E discovers A by Service UUID then writes its payload to A
A discovers E by accepting payload write from E
B discovers F by Service UUID
F discovers B by Service UUID
C discovers G by Service UUID
G discovers C by manufacturer ID then writes payload of D to C
See E to A
See A to E
See E to A
See A to E
F discovers G by Service UUID
G discovers F by Service UUID
See E to A
See A to E

DISCOVERY

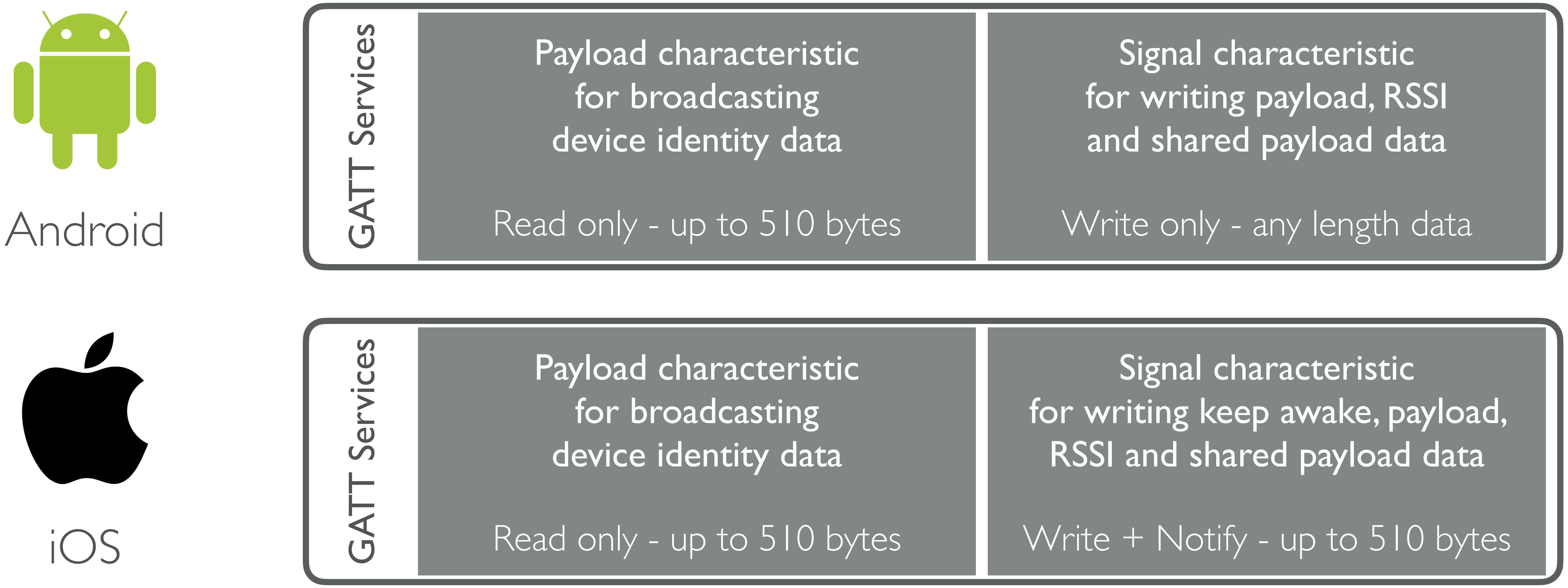
Combining multiple solutions for iOS and Android compatibility

DISCOVERY

- Scan for peripheral by Service UUID works for most cases
  - Android peripheral and central capabilities are identical in foreground and background … **running a foreground service**.
  - Direct detection of Android and iOS foreground peripherals by all Android and iOS centrals.
  - Detection latency in crowded environments determined by device signal strength and radio performance.

- Android phones without advertising support
  - Android central discovers Android and iOS peripheral … **writes its payload to the peripheral**.
  - A pair of non-advertising Android phones cannot discover each other … **chance of scenario in UK is** $(35.2\% \times 53.3\%)^2 = 3.5\,\%$.
  - Scenario has been solved by relaying via another Android or iOS peripheral … **feature disabled as it degrades overall continuity.**

- iOS background peripheral and iOS background central
  - Discovery is possible by enabling (a) location updates, and (b) beacon ranging … advert is read when display is lit even for an instant.
  - Phone display is lit by (a) normal usage, (b) notification, (c) home button press, and (d) raise to wake.
  - Average time between phone pick up during the day is around 10 minutes … **discovery regularly triggered by normal usage**.
  - Peripheral discovers central when central connects to peripheral … **bi-directional discovery when display lit on either phone**.
  - Android central also writes recently discovered iOS peripherals to all iOS peripherals … **discovery via nearby Android central**.



Android

GATT Services

Payload characteristic for broadcasting device identity data

Read only - up to 510 bytes

Signal characteristic for writing payload, RSSI and shared payload data

Write only - any length data

iOS

GATT Services

Payload characteristic for broadcasting device identity data

Read only - up to 510 bytes

Signal characteristic for writing keep awake, payload, RSSI and shared payload data

Write + Notify - up to 510 bytes

SERVICES

Read only payload and multi-purpose signal characteristic

SERVICES

- iOS and Android payload characteristic
  - Read only characteristic for distributing device identity data to iOS and Android centrals.
  - Data encodes identity of physical device for attribution of distance measurements … **enables contact tracing and infection risk estimation**.
  - Payload value generated per request based on current time … **enables mutation and encryption for security, privacy, and mitigates attacks**.
  - Insert your own payload to implement your security and privacy solution … **compatible with centralised and decentralised approaches**.
- Android signal characteristic
  - Write only characteristic for accepting payload, RSSI and shared payloads … **enables full capability for Android centrals that cannot advertise**.
  - Payload signal = Signal type (1 byte) + Payload size (2 bytes) + Payload data (up to 507 bytes)
  - RSSI signal = Signal type (1 byte) + RSSI value (2 bytes)
  - Payload sharing signal = Signal type (1 byte) + RSSI value (2 bytes) + Shared payload size (2 bytes) + Shared payload data (up to 505 bytes)
- iOS signal characteristic
  - Write characteristic identical to Android signal characteristic, except it also supports notification … **enables continuous operation on iOS**.
  - iOS central writes empty data to iOS peripheral signal characteristic to wake iOS peripheral … **prevents iOS peripheral from suspending**.
  - iOS peripheral notifies subscribing iOS centrals of empty data update to wake iOS central … **prevents iOS central from suspending**.
  - All Android centrals share payloads of recently seen iOS and non-advertising Android peripherals … **enables iOS - iOS discovery**.

Measurement target

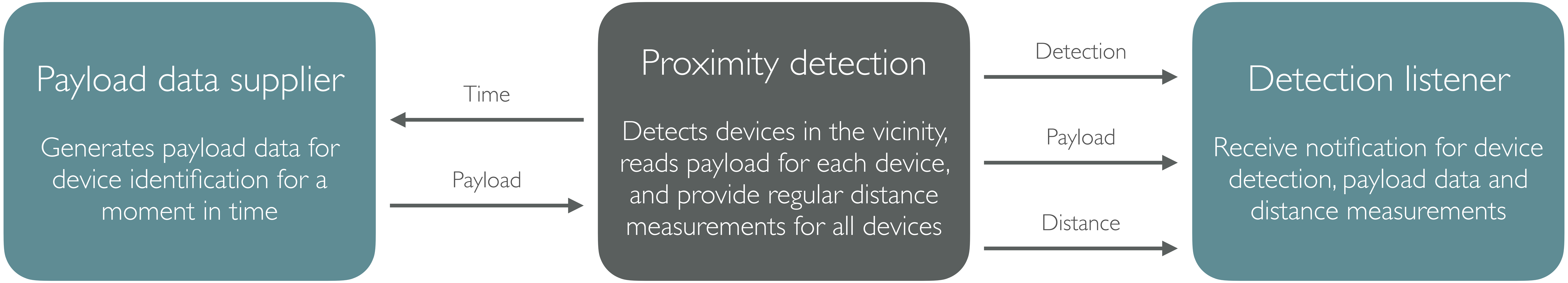
Device measuring distance		
Android	Scan for peripherals at regular intervals to obtain RSSI from peripheral advert discovery data Android measuring Android	Scan for peripherals at regular intervals to obtain RSSI from peripheral advert discovery data Android measuring iOS
iOS	Scan for peripherals at regular intervals to obtain RSSI from peripheral advert discovery data iOS measuring Android	Maintain connection with iOS peripheral to read RSSI value at regular intervals iOS measuring iOS

PROXIMITY

Estimating distance by measuring RSSI

PROXIMITY

- Scan for peripheral at regular intervals works for most cases
  - RSSI value included in scan response for both iOS and Android centrals … **enables fast, low power, and connection free measurements**.
  - Frequent connection to Android devices can silently crash bluetooth capability, reboot required to remedy.
- Android scan
  - Scanning at regular interval requires a reliable and consistent time source on Android phones … **surprisingly hard to achieve**.
  - Android Timer, Handler, and Looper timings drift wildly in background apps, one second delay can often become 30 minutes.
  - Bespoke time source relies on Thread sleep delay loop and Android wake lock … **minimal impact on power usage**.
  - RSSI samples per scan based on advertise mode : low power (few), balanced (several), low latency (many) … **balanced recommended**.
  - Low latency mode offers many samples for more accurate distance estimation but prevents older devices from competing in discovery.
- iOS - iOS measurement
  - iOS central background scan only reports initial discovery of iOS background peripheral … one RSSI measurement only.
  - iOS background app enters suspended state within about 10 seconds of inactivity … no more measurements.
  - Peripheral discovery, connection, and characteristic read / write / notify prevent iOS background app from suspension.
  - Bouncing empty data between iOS peripheral / central via a notifying signal characteristic … **keeps iOS app running indefinitely**.
  - RSSI samples taken at regular intervals over active connection between iOS devices … **enables continuous distance measurement**.



Payload data supplier

Generates payload data for device identification for a moment in time

Proximity detection

Detects devices in the vicinity, reads payload for each device, and provide regular distance measurements for all devices

Detection listener

Receive notification for device detection, payload data and distance measurements

INTEGRATION

Simple and common API across iOS and Android

INTEGRATION

- Proximity detection solution is payload agnostic
  - Implement **PayloadDataSupplier** to use this solution as transport for your existing device identification payload.
  - Implement **SensorDelegate** to receive detection, payload and distance measurement data for your contact tracing and case isolation logic.
- PayloadDataSupplier interface
  - Device identification data encoded into payload data … **encryption and mutation for security and privacy**.
  - Proximity detection solution will request supply of payload data upon read payload request by another device.
  - Proximity detection solution will provide timestamp as basis for encryption and mutation in data generation.
- SensorDelegate interface
  - A set of simple callback functions for receiving proximity detection events and data.
  - Separation of detection and payload … **target may be detected and distance measured before payload is read**.
  - Applicable to both centralised and decentralised approaches.

SUMMARY

- Technical solution derived from epidemiology requirements
  - Works on the vast majority of existing phones on existing software.
  - High performance means efficacy is close to theoretical maximum in the UK.
  - Separation of device identification data enables integration with existing solutions.
- Proposed path for solution adoption
  - Review solution design, test procedure, and test results.
  - Try iOS and Android test apps to gain confidence of capability and claimed results.
  - Review test app code to learn how to integrate the solution yourself …
  - … and/or grant access to your code repository to receive supported integration.