



# Intro to IoT

Igor Efremov





# What is IoT?



## What is IoT?

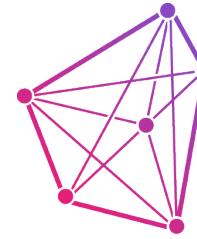
**The Internet of Things is the concept of connecting any device with an on/off switch to the Internet and/or each other.**



- Thermostats
- Light Bulbs
- Door Locks



- Heart Rate Sensors
- Activity Trackers
- Ingestible Sensors



- Vehicles
- Smart Watches
- Home Energy Usage

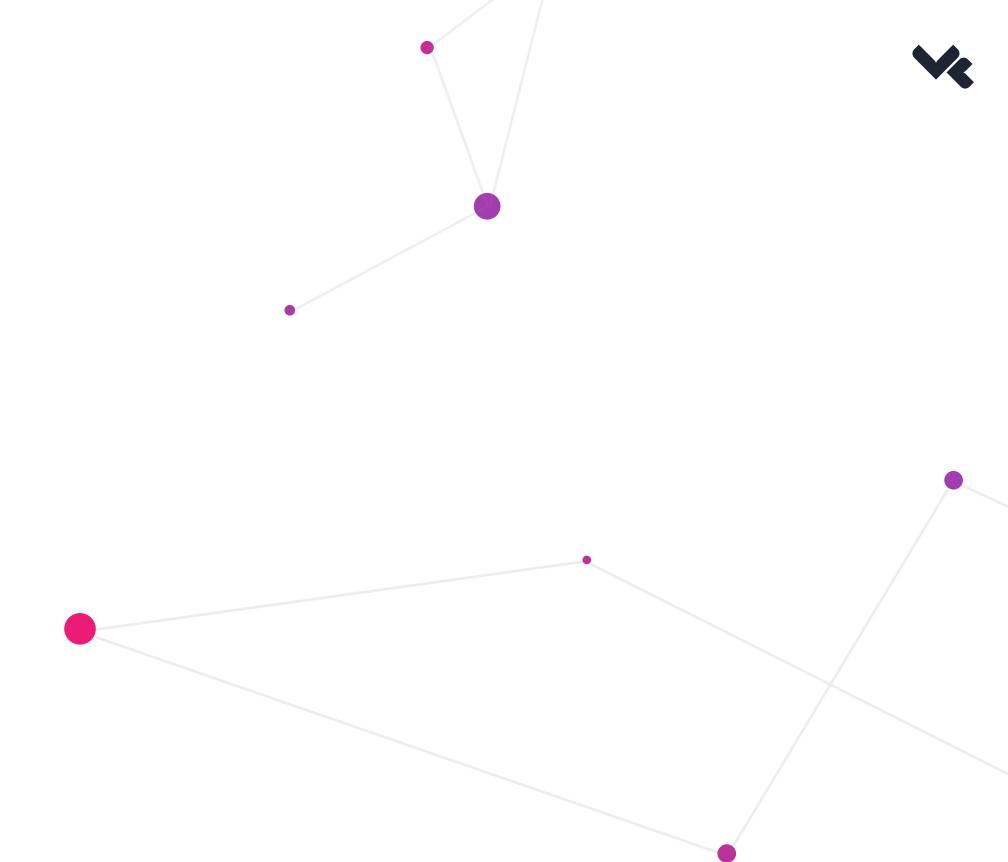


# BLE & Wi-Fi



## Wi-Fi

- Wireless communication method
- Most likely use it every day





## BLE is a version of Bluetooth that was built for IoT.



Ultra-low power consumption

Industry standard

Allows for encryption



## The Generic Access Profile (GAP) controls connections and advertising in BLE.



Defines device roles

Defines advertising and scan response data



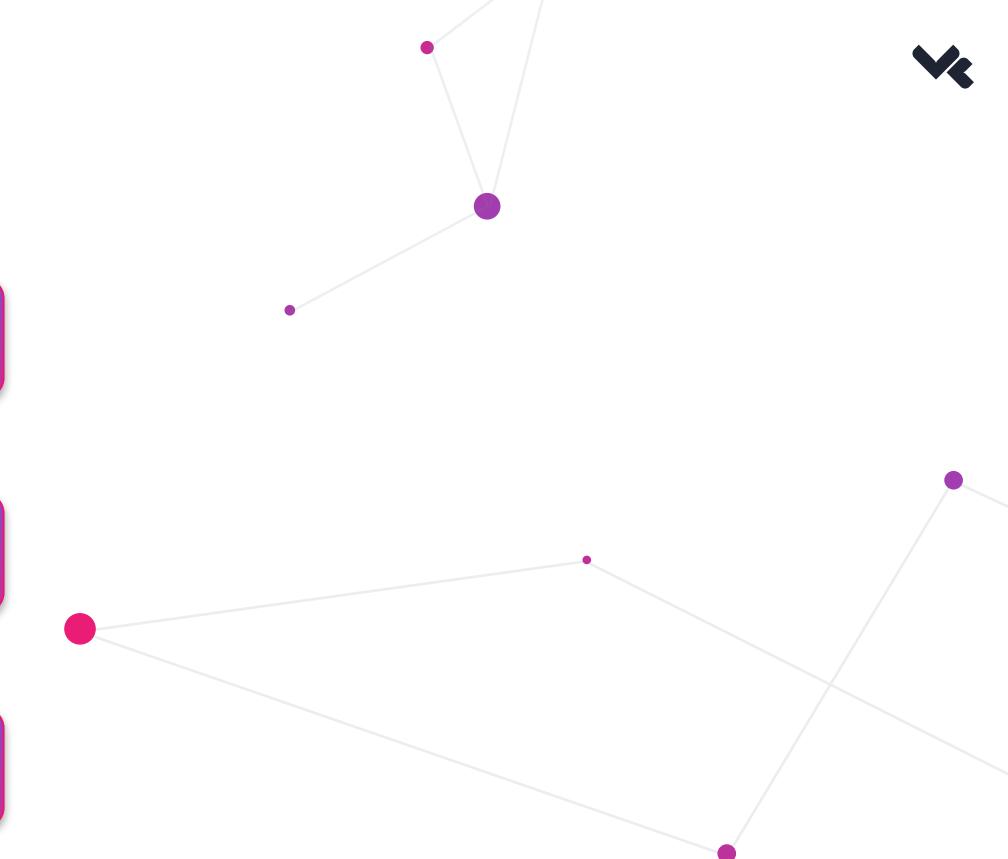
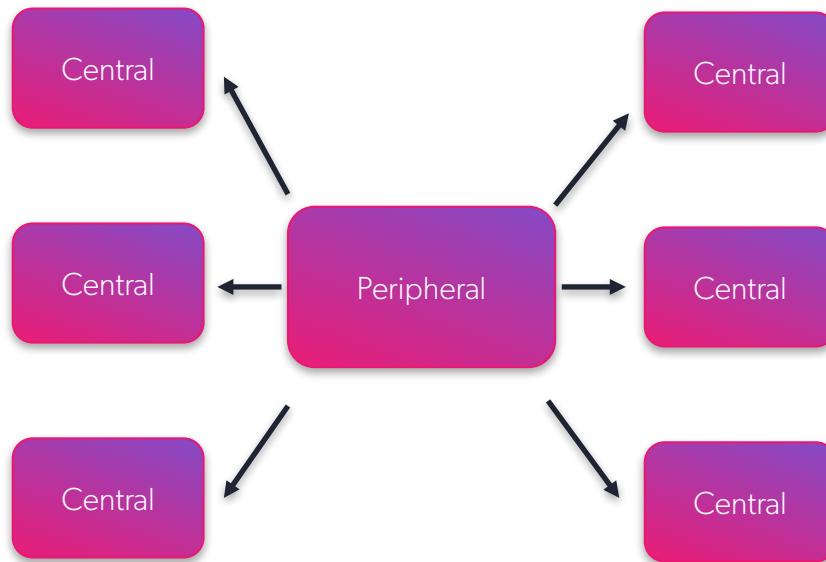
## Device Roles

- Device can be Central or Peripheral role
- Peripheral can advertise
- Central can send a connection request
- i.e. Smart Light Switch is Peripheral and Hub is Central





## Device Roles (cont.)





## Advertising/Scan Response

- Advertising Data is mandatory
- Scan Response is requested by Central (and optional)
- Both payloads are 31 bytes (although 47 are transmitted)
- Scan Response is there to hold extra information (i.e. device name)





## Advertising/Scan Response (cont.)

1 Byte	4 Bytes	2 Bytes	6 Bytes	3 Bytes	28 Bytes	3 Bytes
<b>Preamble</b>	<b>Access Address</b>	<b>PDU Header</b>	<b>AdvA</b>	<b>AD Flags</b>	<b>Advertisement</b>	<b>CRC</b>



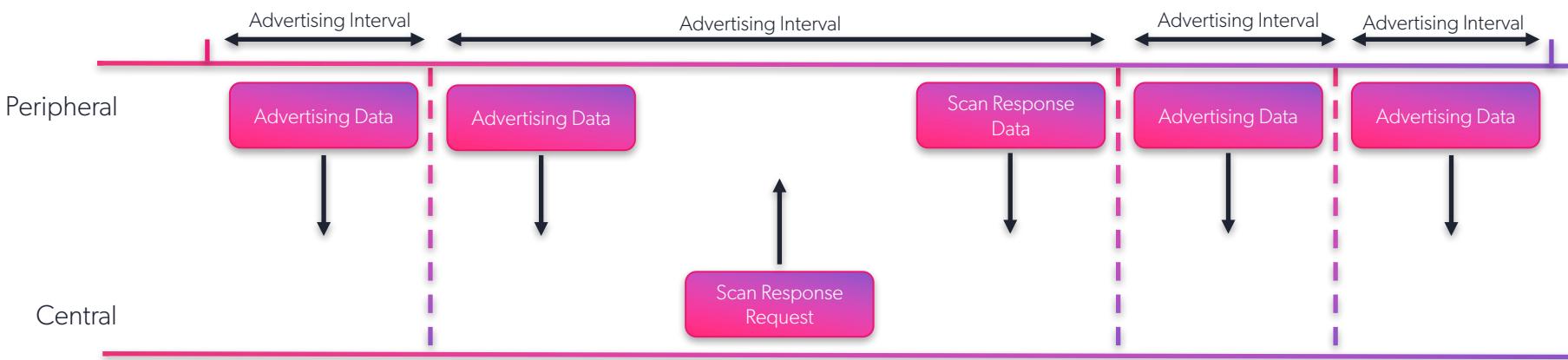
## Advertising/Scan Response (cont.)

- Peripheral sets Advertising Interval
- Every time this interval passes, the main advertising packet is retransmitted
- Longer interval results in lower power usage, but less responsive feeling (i.e. 2 sec vs. 20 ms)
- Again, Scan Response must be requested by Central





## Advertising/Scan Response (cont.)





## The Generic Attribute Profile (GATT) defines the way that two BLE devices transfer data.



GATT comes into play when a connection is established

Advertising (GAP) stops when a connection is made



## GATT

- Connections are exclusive  
(Peripheral can only be connected to one Central at a time)
- Again, advertising (GAP) stops when a connection is made
- Server/Client relationship
- Peripheral is GATT Server (Slave)
- Central is GATT Client (Master)





## GATT Client

- Again, this is the Central
- Also called the Master Device
- Sends requests to the GATT Server  
(Peripheral/Slave Device)
- Starts all transactions





## GATT Client (cont.)

- Upon connection, the Peripheral suggests a Connection Interval
- The Central will try to reconnect every Connection Interval to check for new data, etc.
- Connection Interval is a suggestion from Peripheral (Central may adjust if handling more than one Peripheral)





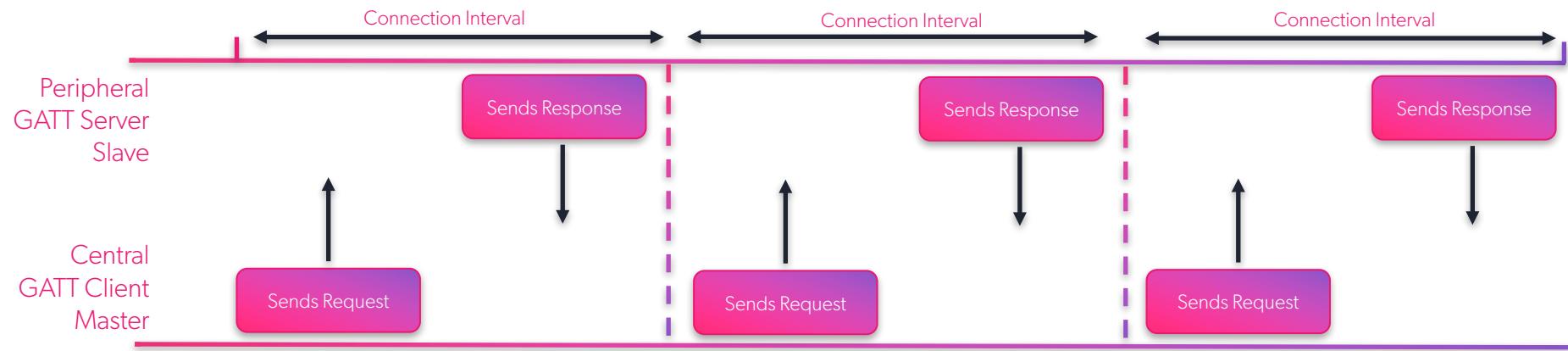
## GATT Server

- Again, this is the Peripheral
- Also called the Slave Device
- Receives requests from the GATT Client (Central/Master Device)
- Can respond to requests from the GATT Client
- Holds Attribute Protocol (ATT) data (Services & Characteristics)





# GATT Transactions





## GATT Concepts

# GATT transactions are based on objects called Profiles, Services, and Characteristics.

### Profiles

- A pre-defined collection of Services that has been compiled by either the Bluetooth SIG or by the developers of the device

### Services

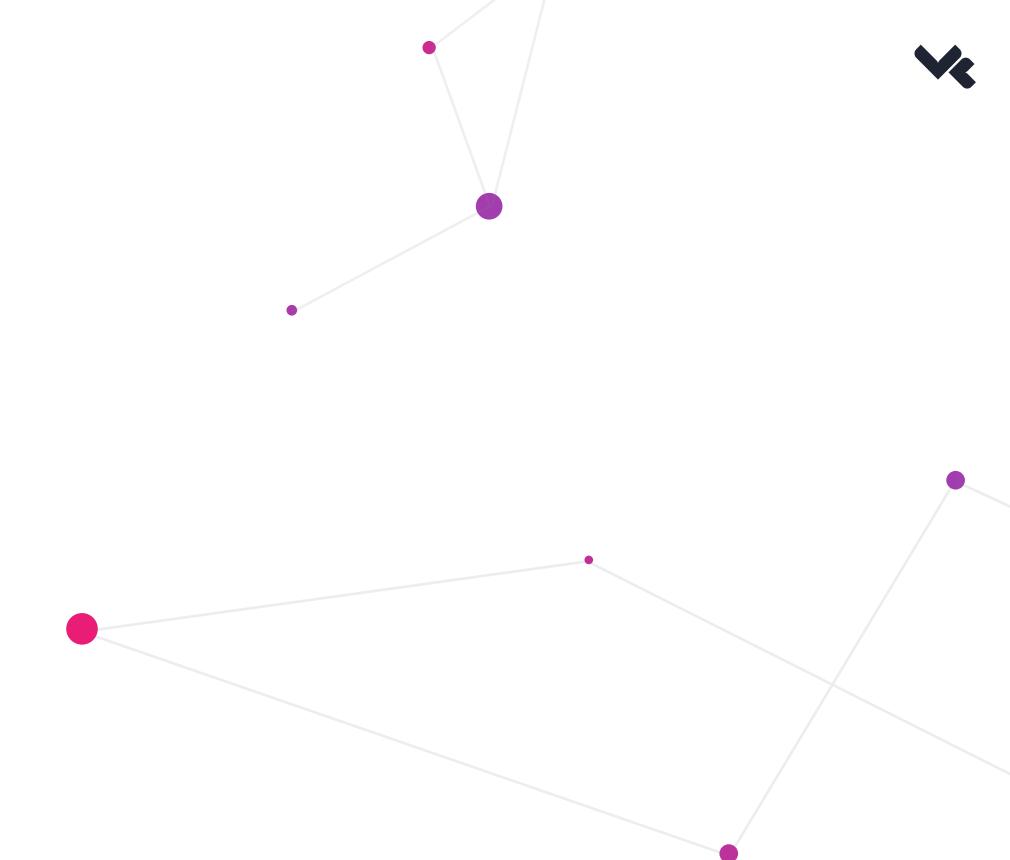
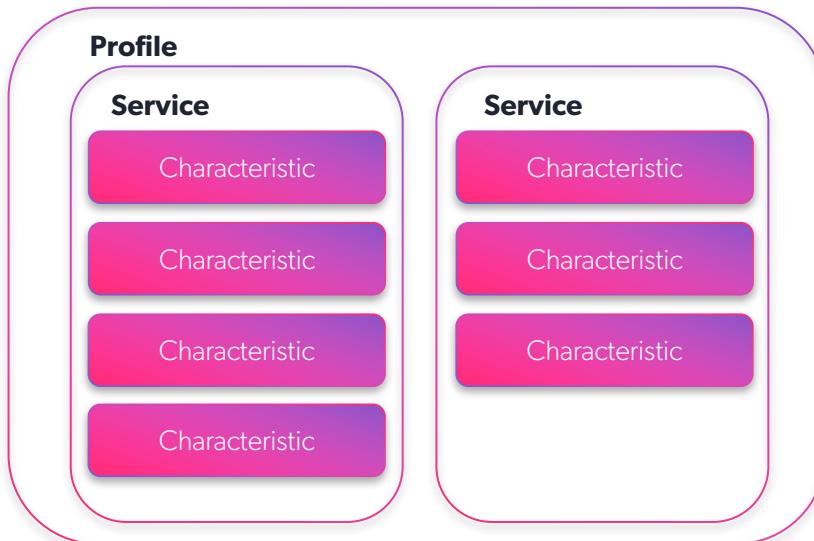
- Used to break up data into logic entities
- Collection of Characteristics

### Characteristics

- Encapsulates a single data point
- A data point can contain an array of related data (i.e. X, Y, Z of accelerometer)



## GATT Concepts (cont.)





## Profile

- Doesn't actually exist on peripheral
- Again, just a defined collection of Services





## Service

- Contains specific chunks of data known as Characteristics
- Can have one or more Characteristics
- Distinguishes itself using a UUID (16-bit if defined by BT SIG; 128-bit otherwise)





## Characteristic

- Main point of interaction on Peripheral
- Can be read and written to
- Peripheral sets read/write permission for each Characteristic
- Distinguishes itself using a UUID (16-bit if defined by BT SIG; 128-bit otherwise)

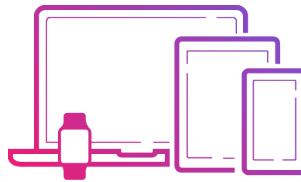




# Implementation



# Implementation



MacBook Pro host computer

Developed using C in Visual Studio Code

Compiled/Flashed with Makefile from Terminal

Nordic nRF52 Development Kit host board during development

Case modeled with Autodesk Fusion 360

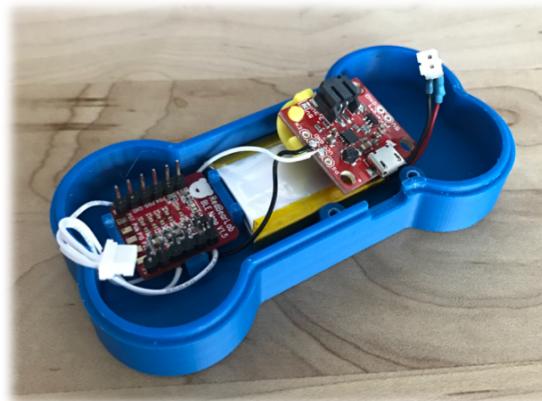
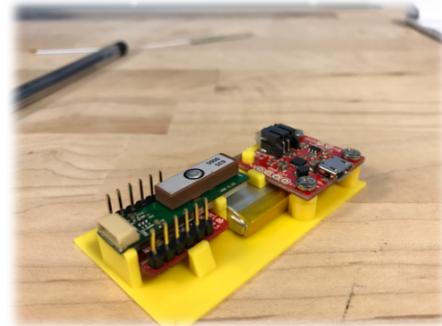
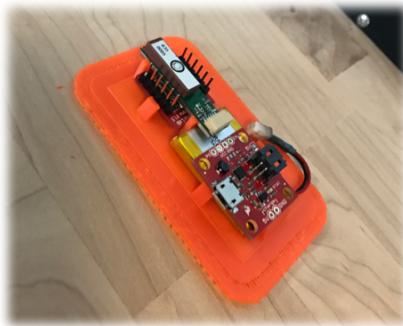
Case printed with Zortrax M200 3D printer

RedBearLab BLE Nano (nRF51822 SOC) used in prototype



# Prototype

# Prototype





# Advertising

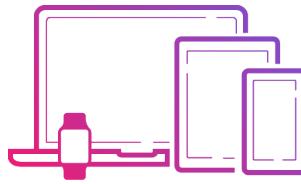
1 Byte	4 Bytes	2 Bytes	6 Bytes	3 Bytes	28 Bytes	3 Bytes
<b>Preamble</b>	<b>Access Address</b>	<b>PDU Header</b>	<b>AdvA</b>	<b>AD Flags</b>	<b>Advertisement</b>	<b>CRC</b>



# iOS (CoreBluetooth)



# Key Concepts



CBCentralManager

CBPeripheralManager

CBCentral

CBPeripheral

CBService

CBCharacteristic

CBUUID

Advertising/Scan Response Data



## CBCentralManager

- Manages discovered or connected peripherals
- Used for scanning, discovering, and connecting
- Essentially is the central





## CBPeripheralManager

- Manages published services and advertises to central devices
- Essentially is the peripheral





- Represents remote central devices that have connected to an app acting as a peripheral





## CBPeripheral

- Represents remote devices that an app acting as a central has discovered advertising or is connected to



## CBService



- Represents a peripheral's service





## CBCharacteristic

- Represents a characteristic in a peripheral's service





- Represents a 128-bit BLE UUID
- Can be initialized from a 16-bit UUID



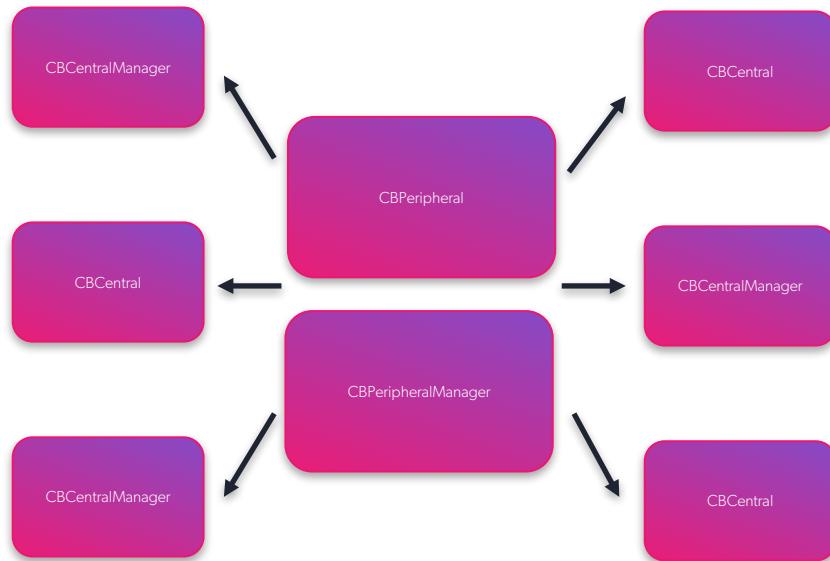


## Advertising/Scan Response Data

- iOS combines advertising and scan response data into one object

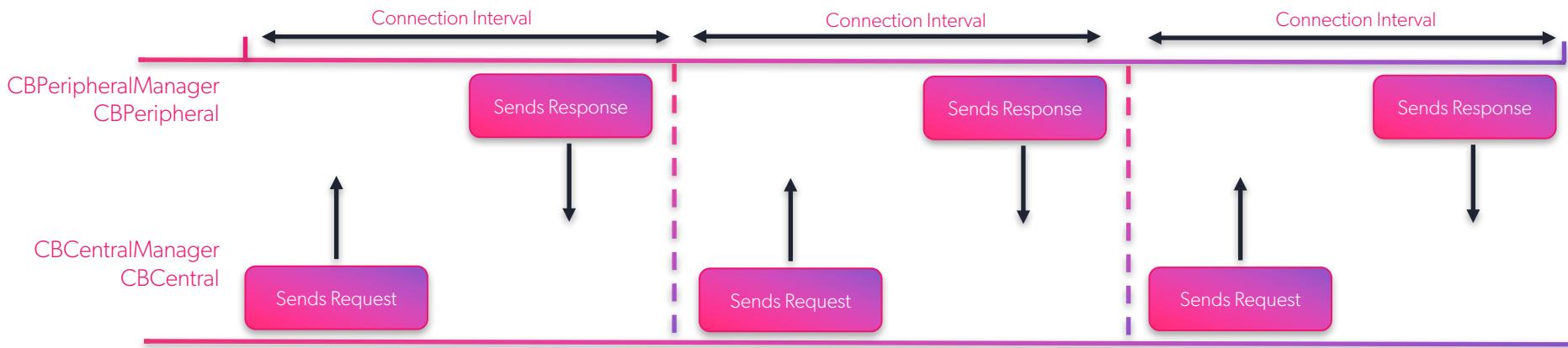


# Device Roles



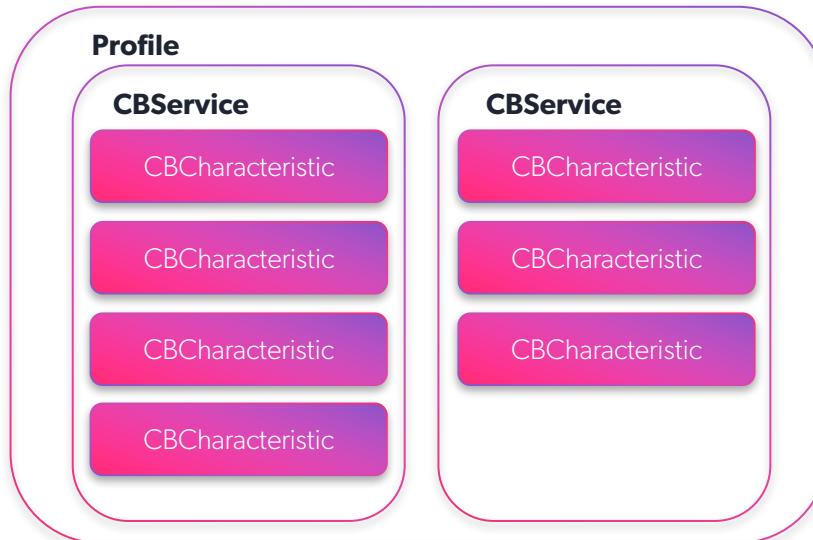


# GATT Transactions





# GATT Concepts





# Closing Notes



## Closing Notes

- BLE is not the only way to communicate
- This was just the surface of BLE
- It's a lot easier than you think





# Resources



## Resources

- <https://developer.bluetooth.org/TechnologyOverview/Pages/Profiles.aspx#GATT>
- <https://developer.bluetooth.org/gatt/services/Pages/ServicesHome.aspx>
- <https://developer.bluetooth.org/gatt/characteristics/Pages/CharacteristicsHome.aspx>
- <https://learn.adafruit.com/introduction-to-bluetooth-low-energy/gap>
- <https://www.nordicsemi.com/eng/Products/Bluetooth-low-energy/nRF52-DK>
- <http://redbearlab.com/blenano/>
- <https://developer.apple.com/reference/corebluetooth>

# Thank You

Vectorform



Invent with us.