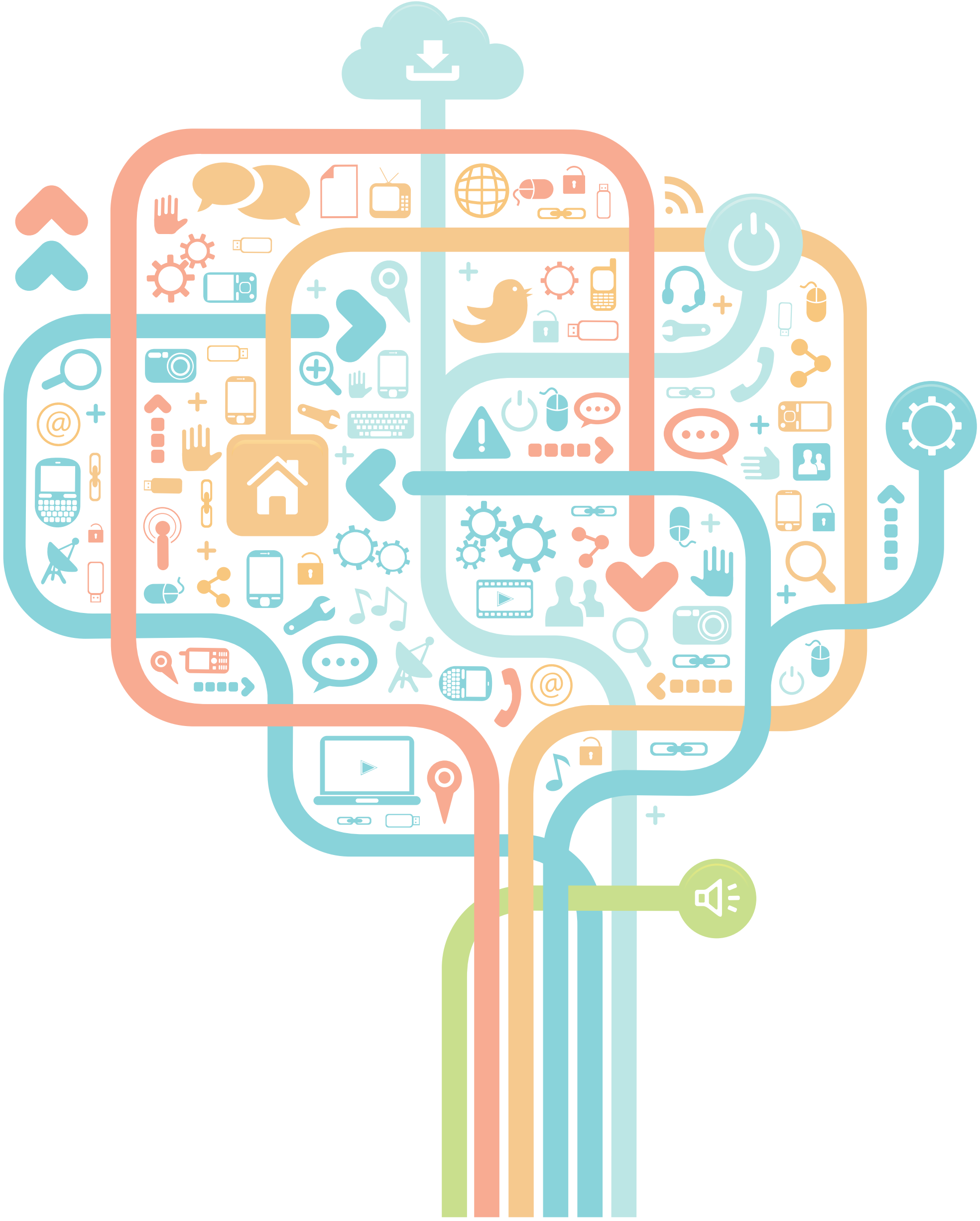
wimoto



WIMOTO MOTES DEVELOPER GUIDE V1.0.3

1 Introduction **3**

1.1 About Wimoto Sensors *3*

1.2 Bluetooth Low Energy *4*
1.2.1 Roles 4
1.2.2 Profiles, Services and Characteristics 7
1.2.3 Notifications and Indications 8

2 Working With Wimoto Sensors **10**

2.1 Development Requirements *10*

2.2 Sensor Specifics *10*
2.2.1 Temperature Primary Service 12
2.2.1.1 Current Temperature 12
2.2.1.2 Temperature Alarm Low Value 12
2.2.1.3 Temperature Alarm High Value 12
2.2.2 Temperature Alarm Set 12
2.2.3 Temperature Alarm Indicator 13
2.2.4 Light Level and Humidity Primary Services 13
2.2.5 Data Logger 13
2.2.6 Device Management Service 14
2.2.7 Device Information Service 14
2.2.8 Battery Service 15
2.2.9 Broadcast (“Sensor”) Mode. 15

3 Further Reading **16**

The information contained in this document is Copyright 2014 Wimoto Technolgoies Inc. and may not be reproduced in part or full without express written consent. All Rights Reserved. Errors and omissions excepted.

If you have feedback, please feel free to contact us at contact@wimoto.com.

1 Introduction

1.1 About Wimoto Sensors

Wimoto sensors are small, battery-powered wireless sensors that monitor various environmental conditions. They communicate using the Bluetooth Low Energy (Bluetooth Smart) standard to modern smartphones, tablets, computer and our own gateway (Mesh).

There are currently five sensors in the Wimoto family. Their names and capabilities are listed below:

	Climate	Grow	Sentry	Thermo	Water
Ambient light	■	■			
Ambient temperature	■	■		■	
Ambient humidity	■				
Soil moisture		■			
Passive infrared motion sensor			■		
Accelerometer			■		
Thermopile/infrared thermometer				■	
Probe thermometer				■	
Water detector					■

The sensors include a lithium coin cell battery (CR2450) and will run for over a year under normal usage conditions. The battery is replaceable by removing the screws on the back of the Wimoto with a Philips PH00 screwdriver.

Things that affect battery life are:

- Being continuously in a ‘connected state’ with the sensor (an unusual use-case as it would require a constantly running smartphone or tablet to be in close proximity to the sensor)
- Aggressive use of the data logger downloading function

Note: even greater power efficiency is achieved when the sensor is running purely in Sensor (Broadcast) mode. See below for an explanation.

The range of the sensors is highly dependent on physical obstructions with metal and concrete being among the least desirable materials. However, a range of 35m/100ft in open air is achievable but is also partly dependent on the device making the connection.

1.2 Bluetooth Low Energy

Bluetooth Low Energy was introduced in the Bluetooth Core Specification version 4.0. It should be noted that Low Energy is an optional component of the Core Specification and not all Bluetooth 4.0 devices support Low Energy. To further confuse matters, Bluetooth Low Energy is marketed as “Bluetooth Smart”.

For a Bluetooth 4.0 host controller to work with Wimoto sensors, the host controller should be either “Dual Mode” capable (meaning it supports both classic mode Bluetooth and Bluetooth Low Energy) or a “Single Mode” Bluetooth Low Energy device (meaning it only supports the Low Energy protocol).

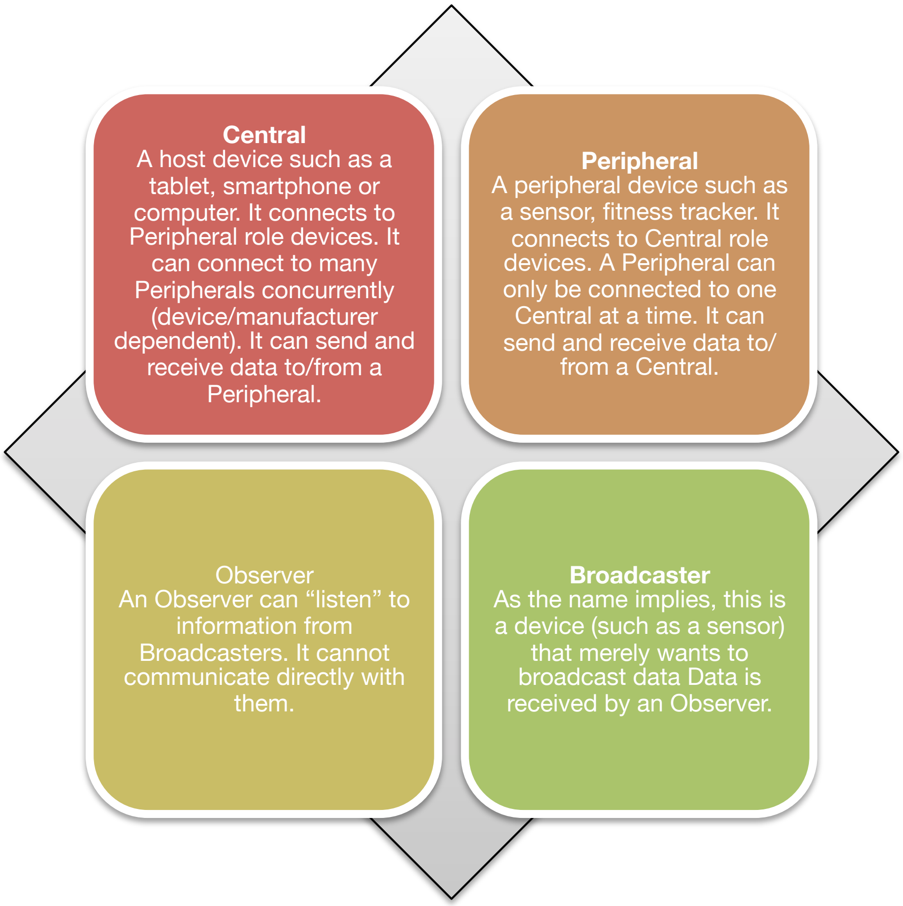
From this point on, we will refer to Bluetooth Low Energy by the acronym “BLE”.

1.2.1 Roles

Bluetooth Low Energy has a master-slave relationship at both the communications protocol level and the data access level. For now, we will focus on the communications protocol layer which is called the Generic Access Protocol (GAP).

GAP supports a number of “roles” and a Bluetooth Low Energy device is allowed to implement a minimum of one (1) role and may in fact implement them all.

The currently defined roles are in the chart below.



Central
A host device such as a tablet, smartphone or computer. It connects to Peripheral role devices. It can connect to many Peripherals concurrently (device/manufacturer dependent). It can send and receive data to/from a Peripheral.

Peripheral
A peripheral device such as a sensor, fitness tracker. It connects to Central role devices. A Peripheral can only be connected to one Central at a time. It can send and receive data to/from a Central.

Observer
An Observer can “listen” to information from Broadcasters. It cannot communicate directly with them.

Broadcaster
As the name implies, this is a device (such as a sensor) that merely wants to broadcast data Data is received by an Observer.

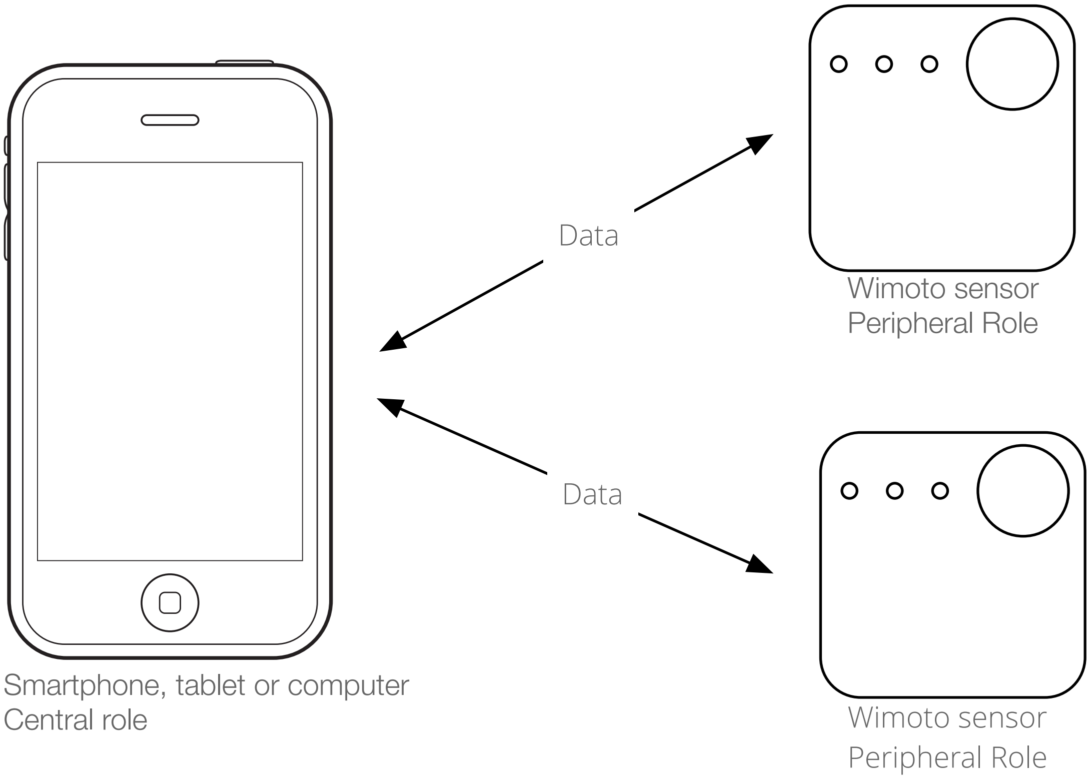


Figure 1: Central-Peripheral role

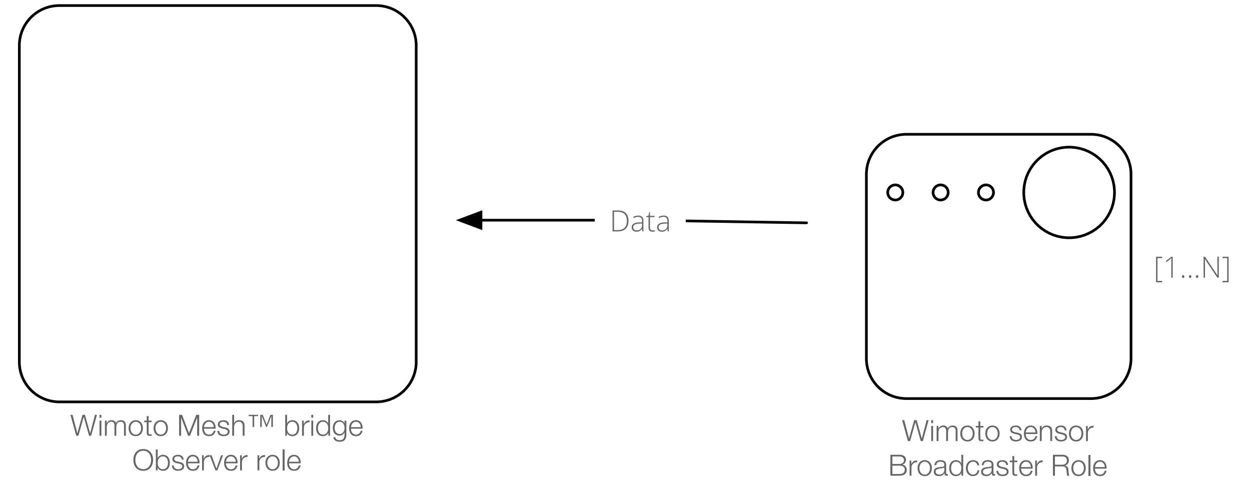


Figure 2: Observer-Broadcaster role

1.2.2 Profiles, Services and Characteristics

Bluetooth Low Energy implements access to data in an object-oriented fashion. A BLE device (a Peripheral or Broadcaster) implements one or more “Services” that contain one or more data characteristics (Characteristics). A collection of Services and Characteristics is referred to as a Profile. A blood glucose meter, for example, might have a profile that includes a current glucose reading Service and Characteristic, a Battery Level Service, and a Device Information Service with manufacturer information such as serial number and model.

The object-oriented nature of BLE means that a Service can also include other Services as sub-services, which in turn have their own data Characteristics. This is referred to as a “Secondary” rather than “Primary” Service.

Characteristics can be self-describing by having descriptors attached to them. The Bluetooth Special Interest Group (Bluetooth SIG) manages these descriptors.

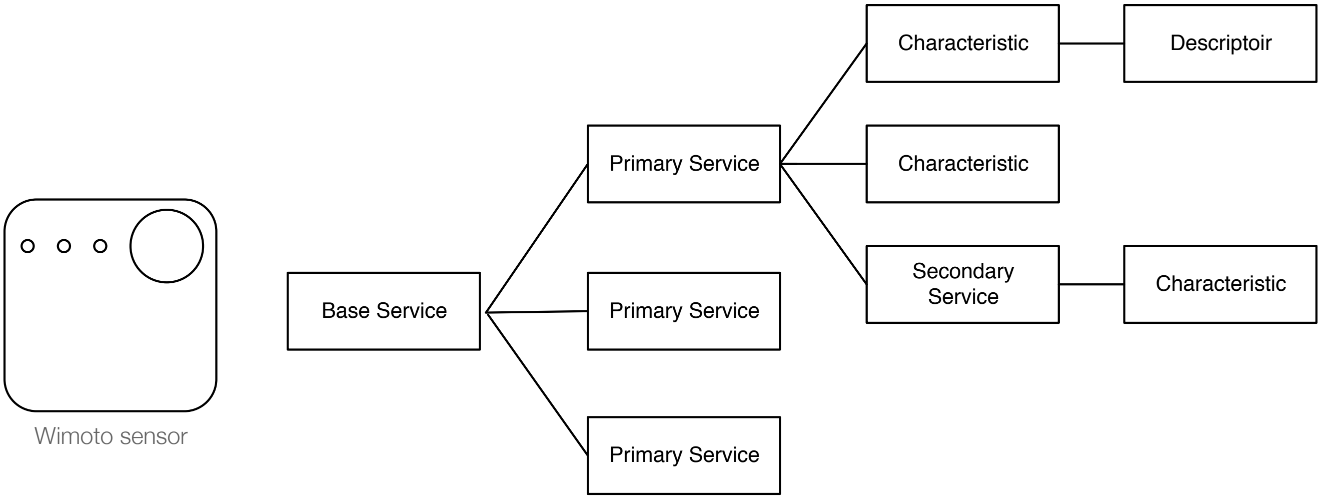


Figure 3: Relationship between Services and Characteristics

Services and Characteristics have unique hexadecimal identifiers (UUIDs). The Bluetooth SIG has defined some common Profiles and they carry shorter 16-bit (four hexadecimal characters) UUIDs. Custom Services and Characteristics carry long 128-bit UUIDs which are made up of a identifier unique to an organization plus 16-bits specific to the Service or Characteristic it’s applied to. A manufacturer may additionally have a Company Identifier – Wimoto’s is 0x0107 -- whoch

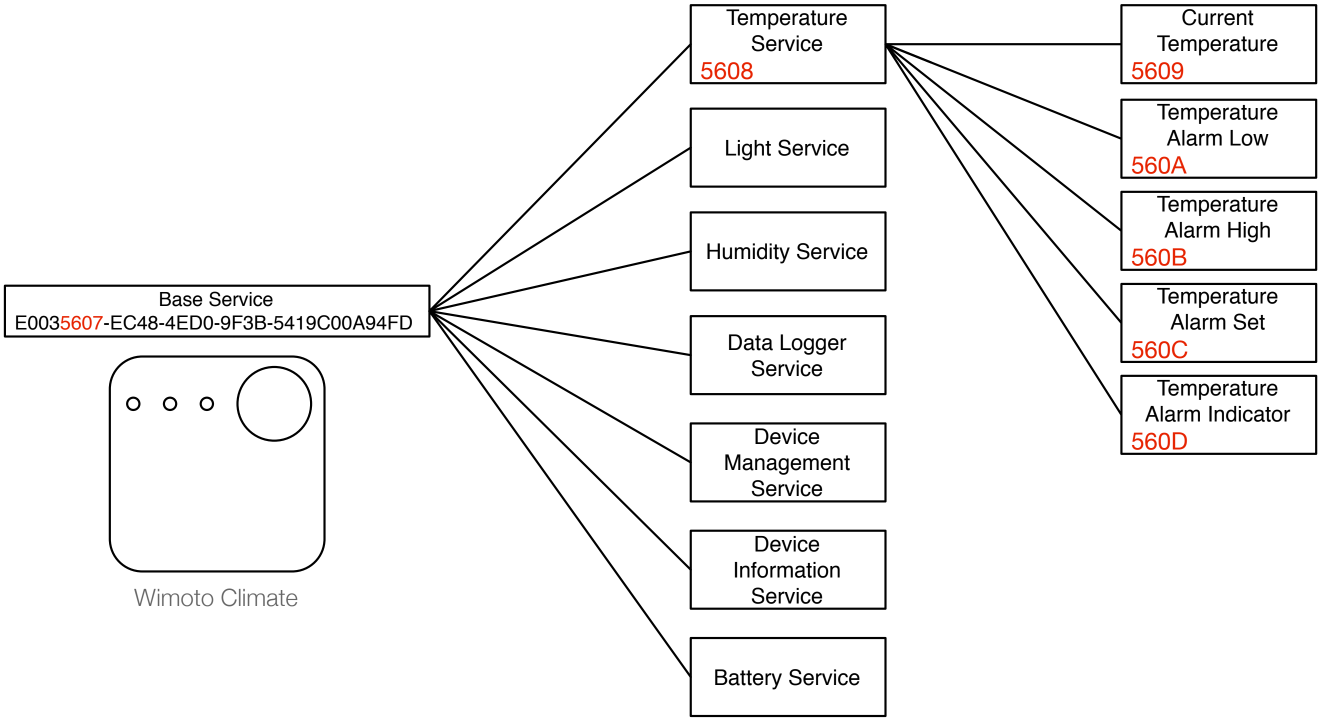


Figure 4: Partial map of Wimoto Climate profile

1.2.3 Notifications and Indications

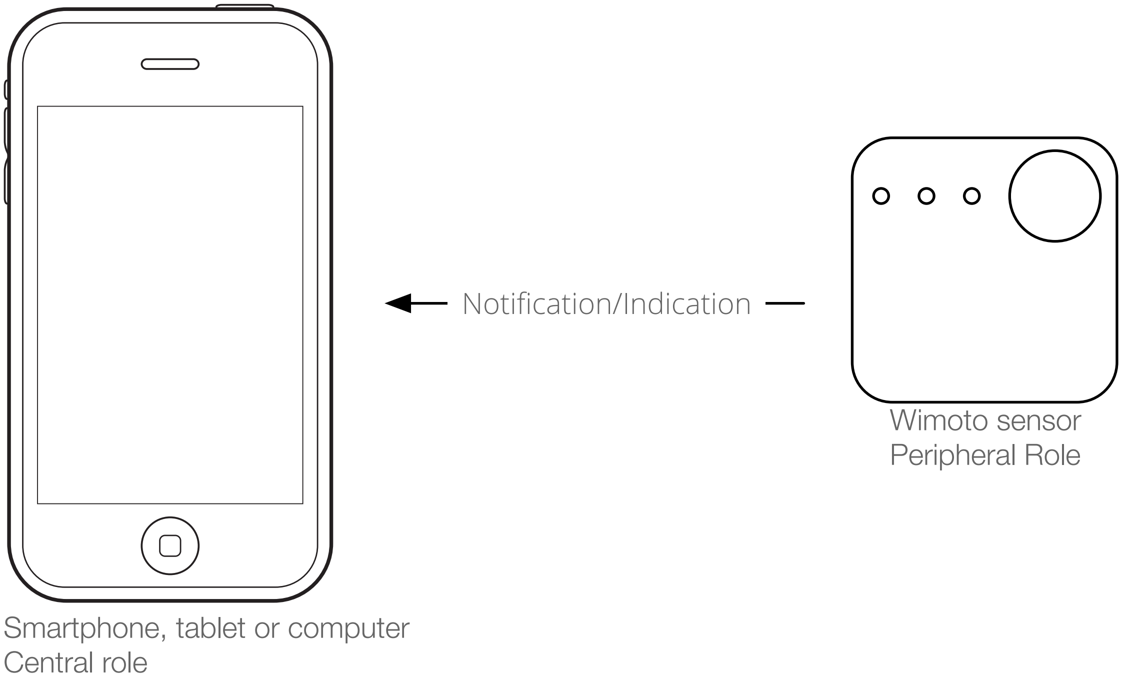


Figure 5: Central's 'subscribe' to Notifications and these are 'pushed' from a Peripheral asynchronously.

A BLE Central can "subscribe" to Notifications and Indications from a Peripheral (this is not available in the Observer-Broadcaster relationship) and have updates 'pushed' to the Central rather than continuously polling for changes. This is an extremely powerful feature and Wimoto takes advantage of this for things such as data updates, alarms, and transferring large amounts of data for the data-logger.

For example, a Central may subscribe to the current temperature characteristic on a Climate and then get a notification every time that temperature value changes. This happens completely asynchronously.

An Indication is similar to a Notification except the Central has to specifically respond to an Indication. Wimoto uses this for alarms.

2 Working With Wimoto Sensors

2.1 Development Requirements

To communicate with Wimoto Sensors, you will need:

- Hardware that supports Bluetooth Low Energy (referred to as “Bluetooth Smart Ready”). Many current generation smartphones and tablets support BLE – all Apple iPhones after the 4s, all iPads after 3rd generation, many Samsung, HTC and Google Nexus devices, most newer Macs.
- Access to the “Bluetooth stack” via the programming language of your choice.

iOS and OSX: Apple’s SDK/APIs feature something called “CoreBluetooth” which is available in iOS > 6 and OSX > Lion.

Android: Android 4.3/Developer API 18 introduced Google-provided BLE APIs and a stack. Previously, manufacturers such as Samsung implemented their own, which caused app development challenges.

Linux: Linux kernel > 3.6 features the BlueZ Bluetooth Stack which has good support for Bluetooth Low Energy. Version > 4.99 of the BlueZ utilities is required to make best use of this environment.

Embedded: consult your silicon vendor as to the availability of a suitable BLE stack or BLE radio modules that extrapolate the BLE stack.

2.2 Sensor Specifics

We’re going to use the Climate as a use-case, but this applies equally to all of our sensors.

Figure 6. shows the full map of the Climate’s Profile. Our private UUIDs have been shortened from 128-bits to 16-bits for clarity. Please refer to the Excel spreadsheet “Wimoto Profiles” for full UUIDs.