Beacons: The Technical Overview

This paper seeks to provide education and technical insight to beacons, in addition to providing insight to Apple's iBeacon specification.
## Contents

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objective</td>
<td>3</td>
</tr>
<tr>
<td>Audience</td>
<td>3</td>
</tr>
<tr>
<td>Bluetooth® Low Energy overview</td>
<td>3</td>
</tr>
<tr>
<td>Types of BLE modules</td>
<td>4</td>
</tr>
<tr>
<td>Design considerations</td>
<td>5</td>
</tr>
<tr>
<td>Button-cell batteries</td>
<td>6</td>
</tr>
<tr>
<td>BLE iBeacon</td>
<td>7</td>
</tr>
<tr>
<td>Configuration</td>
<td>8</td>
</tr>
<tr>
<td>Battery health</td>
<td>8</td>
</tr>
<tr>
<td>Proximity aware</td>
<td>9</td>
</tr>
<tr>
<td>iBeacon functionality</td>
<td>10</td>
</tr>
<tr>
<td>Conclusion</td>
<td>12</td>
</tr>
</tbody>
</table>
Beacons: Technical Overview

Objective

The objective of this white paper is to provide education on the core functionality and specification of Bluetooth® Low Energy (BLE). The paper will explore how this relates to the creation of BLE beacons, or iBeacons as they are commonly known following the system wide support Apple introduced with iOS 7. The paper describes the capability of iOS 7 and what this enables from an application and system perspective.

Audience

This paper is aimed at the more technologically knowledgeable reader, although it is not intended to exclude the general reader.

Bluetooth® Low Energy Overview

Bluetooth® Low Energy (BLE), otherwise known as Bluetooth® Smart, was created as a way to greatly reduce the amount of power required to connect devices and, in turn, reduce battery consumption. In many ways BLE is separate to Bluetooth because the aims and objectives, as well as the power usage of the technology, differ greatly.

Bluetooth® has found its place in streaming audio and transferring large data files through pairing with devices such as smartphones and computers. BLE, however, is a key enabler for the Internet of Things to connect smart devices like wearables for example. This is why it is the perfect technology for underpinning beacons. BLE can be used in situations where connection times and data sent is minimal, resulting in longer battery life.

Devices, such as wearables and beacons, that connect or interact with a smart device via BLE, see virtually no resultant power drain. Compared with Bluetooth®, BLE is 50-90% more energy efficient, greatly improving the battery life of phones and tablets.

In The Developers Handbook, Robin Heydon describes how the design approach of BLE has taken a completely different direction to Bluetooth®. Heydon compares the approach of Bluetooth® to other wireless methods, which have strived for faster data speeds for communication. BLE, on the other hand, has been optimised for low power. This has been achieved by reducing data rates and connection times. BLE also addresses the cost implications of high volume deployment, with the intention of providing devices, objects and spaces with a means of connectivity.

A fundamental design requirement for cost effective BLE devices lies in the choice of battery. In the case of beacons, this can be done using button-cell batteries that last for up to two years.
Beacons: Technical Overview

Types of BLE Modules

There are essentially three types of Bluetooth® modules that can be incorporated into an overall design:

- Bluetooth®
- Single-mode BLE
- Dual-mode BLE

Bluetooth® is what many people are already familiar with. As already touched upon, Bluetooth® is generally used to either stream audio, found in headsets or speakers, or to transfer large data files between devices, such as AirDrop. Many people perceive Bluetooth® to be battery intensive and so many people have it turned off.

This is no longer necessarily the case. When Bluetooth® 4 was introduced in 2010 with the commercial launch of the iPhone 4S, bringing vast improvements in the power efficiency of Bluetooth®. More importantly, Bluetooth® 4.0 included BLE technology, opening up new opportunities for connected wearables and sensors.

Single-mode modules (otherwise known as Bluetooth® Smart) are purely BLE. These include sensor devices, keys, health monitors and beacons. Single-mode modules can talk to devices featuring dual-mode modules (otherwise known as Bluetooth® Smart Ready), such as smartphones, tablets or computers. Single-mode modules can’t talk to devices that use Bluetooth® modules older than Bluetooth® 4.0, such as phones, wireless headsets or printers.

In the case of standalone beacons, the BLE type most commonly used is single-mode. Where it gets really interesting, is when you consider that all iOS devices sold since the iPhone 4S (over 250 million devices) include Bluetooth® 4.0. This demonstrates the potential to utilise dual-mode BLE in post 2010 iOS devices to send and receive beacon signals.
Beacons: Technical Overview

Design Considerations

Designed by the Bluetooth® Special Interest Group (SIG) the output of BLE is robust, interoperable and high quality.

BLE incorporates “adaptive frequency hopping” enabling beacons and other BLE enabled technology to avoid interference on the 2.4Ghz band, also used by Bluetooth®. The 2.4Ghz band is unlicensed, meaning there are no charges for using it. By using this license free band, BLE can be used anywhere around the world, making deployment of beacons much easier than many other wireless technologies.

Range is the next major consideration for delivering power efficiency. BLE can be set to transmit at different strengths, generally optimised for the type of deployment being carried out. The further the range, the more power is being utilised. BLE has a maximum range of 50m, though the majority of implementations of the technology tend to be a couple of meters. In the case of beacons, we generally find that 30 meters is the optimum furthest range without degrading battery efficiency too much.

The frequency of transmission is another variable that impacts power usage. BLE allows signals to be broadcast as frequently as every 10ms. This, however, would lead to a decline in power efficiency and isn’t really needed for the vast majority of implementations of the technology. A more appropriate frequency of 200ms and above leads to the perfect mixture of battery efficiency and use case implementation. When it comes to iOS, the BLE refresh rate support is set at every second, creating the optimum frequency for both battery life and user experience.

To summarise:

- BLE is specifically designed for low power consumption.
- The design of apps that utilise BLE and beacons must consider the variables, transmission strength and transmission frequency, in the specific use case.
- BLE is not a replacement, or an optimisation of Bluetooth®. The two should not be mistakenly interchanged, they serve different objectives.
- BLE is aimed at market sectors that involve broadcasting snippets of data relatively infrequently. Beacons are one example of this technology.
Beacons: Technical Overview

Button-cell Batteries

One of the advantages of beacons is that they can run on battery power. The batteries generally used for powering beacons are the same type one might find in a watch, Button-cell batteries. They are relatively small, thin and efficient, making them ideal for beacon deployments. Power for beacons can also be hardwired into existing equipment (with customisation); but for deployments that lack a direct power source, button-cell batteries do the job.

Button-cell batteries will typically be able to harness around 80% of their total capacity, especially if they are deployed in low temperature environments. Battery output capacity is generally impacted by environmental temperature, internal currency leakage and how much it is used. This all needs to be taken into consideration when deploying beacons, as larger batteries may be required.

Having researched the market, we came to the conclusion that the CR2032, typically found in beacons, wouldn’t suit the needs for the majority of beacon use case deployments. This is mainly as we don’t believe provides the right balance of longevity to provide a consistently good user experience. As a result, we have opted for a higher capacity button-cell battery that, whilst thicker, will be more reliable and longer lasting over time.

We know from experience that Moore’s Law means every two years, the number of transistors on integrated circuits doubles. As a result, beacons being deployed now will likely be replaced in two years. As low cost wireless devices, this is certainly a further advantage of utilising a battery-powered beacons.

To summarise:

- Beacons run on button-cell batteries; a larger battery provides a more robust user experience with longer battery life.
- Some beacons being deployed now may well be replaced in two years when the batteries need to be replaced, as the modules will be smaller and more advanced.
Beacons:
Technical Overview

BLE iBeacon

As mentioned in the introduction, iBeacon is an Apple trademark for its implementation of BLE beacons. iBeacon is a set of specifications, introduced by Apple, for modules that wish to use the iBeacon branding. The specifications also cover how to ensure that devices running iOS7.x are able to understand the signals being transmitted. Despite being developed by Apple, iBeacons should be able to work on other platforms (where support for BLE is included and libraries are used by apps to understand the signals).

The specification is intended for developers, so this paper will not explore it in any detail. The most important information to understand is that iBeacons broadcast three ID’s, used to create the context in which the iBeacon is placed. Most significantly, there is no connection made with iBeacons to transfer data. iBeacons simply broadcast their presence, with the rest being done by the app.

As a result of this, it would be possible to have a hundred iBeacons in the same location without experiencing any interference. No connection is being made so the device can’t get confused. A device will only take an action if an app recognises one of the broadcasts from the corresponding iBeacon (this will be explained in the proximity aware section on page 10).
Beacons: Technical Overview

Configuration

iBeacon ID's need to be configured for specific apps. There are a few ways to do this:

1. Arrange during the manufacturing process. This has its limitations, particularly in terms of quality control and future proofing.

2. Use a dedicated application to configure the devices. This is our preferred method. This is because it provides better handle on quality control, by being able to adapt functionality as a result of any unforeseen circumstances.

Once configured, it is unlikely they would need to be configured again. The values don’t have to be read-only and can be written too. The other parameters that can be written are the transmission strength and frequency. This flexibility allows for a range of different deployments and new use cases.

Battery Health

Battery health is key for iBeacon deployments, especially in environments that could detriment the battery. Despite the prolonged battery life that BLE provides iBeacons, monitoring battery health provides assurance when deploying and managing the technology.

To monitor battery health, a connection must be made to access the battery status. As the connection takes place in under 3ms, there is minimal affect on battery longevity. Doing so will not interfere with the user experience, as the iBeacon is able to continuously broadcast whilst sending battery health data. Battery status is something that can be monitored by devices coming into range at set parameters; such as daily or weekly rather than every time. This will differ from deployment to deployment so will ultimately depend on the use case.

When managing iBeacons, algorithms can be built into management systems that can alert when batteries need to be checked if no devices come into range for a long period of time. This means that iBeacons can be easily managed.
Beacons: Technical Overview

Proximity Aware

iBeacons provide proximity awareness to objects. This means smart devices with Bluetooth® 4.0, and support for BLE, can determine the proximity to an iBeacon by receiving the broadcast. This is done by the device operating system or dedicated libraries. The proximities are derived using the relative signal strength indicator (RSSI) that outputs at three tiers of proximity:

1. Immediate
2. Near
3. Far
4. Out of range (labelled as unknown)

These proximities are relative to the set transmission strength, i.e. the stronger the signal strength, the further the levels of proximity will be. This provides another level of control within use cases. Each of these proximity tiers can be utilised to activate a different action through an app.

Entering a trade event, an iBeacon set at the entrance door would trigger a ‘welcome to the event’ notification on the device as you approach the event (far range). If you unlocked your device, the app may display the welcome screen with estimated queuing time, or even information about where a shorter queue might be. As you get closer, the app would know how many of your contacts are already inside (near), information which you could access if you chose to get your device out. By moving into the immediate range, the app would automatically log you into the event. All of this could take place without you getting your phone out, helping to speed up the entry to the event and improve the overall experience.

Despite iBeacons being proximity based, triangulation/trilateration is not yet proven or, for those making progress in this area, not yet robust enough to deploy. This is due, in part, to multi-path effects (where the signal travels further to get to the receiver than they need to) resulting in inaccurate calculations of the distance to an iBeacon. It is this same effect that can affect GPS making it inaccurate when indoors, or on cloudy days.

At present, with iBeacons, there is too much error in the proximity calculations to accurately derive position. Additionally, some environments where iBeacons will be deployed make it difficult to come up with a one-size-fits-all solution to the problem. If it is deemed a matter of great importance to have accurate distance for a particular use case, this could be implemented at the back-end, or by utilising other wireless signals and sensors; though this will likely be a bespoke solution.
Beacons: Technical Overview

iBeacon Functionality

The intention of this paper is not to explore the endless opportunities and possibilities of iBeacons, but more to consider the implementations from a technical perspective. As discussed, the iBeacon broadcasts its presence and an app will use it to determine the users proximity from the iBeacon. Coupled with the proximity, the broadcast becomes a sophisticated triggering mechanism.

In a simple example consisting of a single iBeacon, the objective is for the application to surface content that relates to this iBeacon's placement (as already mentioned in the trade exhibition example). The application must firstly identify whether the iBeacon is associated to it and, if it is, actions can be set up in two ways.

Firstly, iBeacons can trigger an action in an app, such as displaying a specific menu, notification or content. This version highlights one of the key advantages of an app; offline capabilities. This type of deployment will be most successful for environments where mobile data or Wi-Fi access is limited.

Alternatively, iBeacons can trigger an action via a cloud-based content management system (CMS). These types of deployments will generally be where the user always has good connectivity, such as in an office, the home, or increasingly, retail environments. As with many types of apps, real life deployments will see a blending of the two methods; where content is downloaded to a device to be stored offline and accessed when in the right location.

In the recent iOS 7.1 update, an app does not have to be ‘running’ in the background to be notified about the user entering or exiting iBeacon regions. This means that as long as the right app is installed on the device, an iBeacon broadcast is still able to work.

As mentioned, each iBeacon proximity region can activate a specific action within an app, though other mechanisms can also be used. These include building in intelligence that will only activate an action if, for example, a device is in a set region for a preselected period of time. Alternatively, it could be set to activate during specific time periods, such as breakfast options in the canteen between 7-10am. Furthermore, the number of visits to a region can also be used to trigger content that recognises the number of times the user has been in the region.
Beacons: Technical Overview

When there are multiple points of interest (POI), such as objects in close proximity to each other, things can get more difficult. Whilst a possible option might be to set the trigger to the immediate proximity zone, it won’t create a good user experience. The user would be required to move their phone near to each object to distinguish which object it is closest to. Instead, it might be that the iBeacon is used to identify the zone and bring up a QR code reader in the app, to bring up information on each object. Alternatively, a single iBeacon would know the objects nearby and let the user select between them.

Deployment and design strategies will play an important role in the adoption of iBeacons. The user journey and the expected functionality must be carefully considered from the perspective of the user. Doing so will maximise the adoption and lead to positive experiences with iBeacon technology. From our research and development, including trial projects, switching content at every beacon is not always possible, practical or in fact desirable.

The premise of iBeacon is to engage the user, provide them with what they actually want, when they want it. Not adhering to this will kill the adoption of iBeacon technology. There needs to be greater intelligence applied to the integration of the technology, both from a strategy and technical perspective. A more subtle approach to using iBeacons may be the most valuable.

The primary channel for iBeacons has been touted to be in the retail space; though there is a big danger that over zealous companies will spam users with offers and notifications that ruins the user experience. As a consequence, users may become distrustful of the technology and consider it only as a commercial sales tool. iBeacons are about trying to think of a way to provide the best user experience possible. By getting the user experience correct, companies will benefit from a more engaged audience, that want to use and interact with their brand in the real world.

To summarise:

- iBeacons trigger actions in apps, with content either stored in the app or from a cloud based CMS.
- A single iBeacon can trigger three actions based on the proximity of a device, which is either within immediate, near or far proximity regions.
- Intelligence needs to be built into apps to only trigger content at the right time, iBeacons can help the app to register the number of times a user enters its region and how long they remain in the region.
- iBeacons are about delivering a better user experience that is helpful, not invasive and contextual to the user needs.
Conclusion

While this is an overview and perhaps a reasonably in depth overview, there are more technicalities to BLE. Already, iBeacon support in iOS is in its second iteration, which indicates more enhanced functionality will come with future versions of the OS.

Apps are where the magic happens. The use cases are endless, it just has to make sense and add value to the user for adoption to occur. Fortunately, we are seeing more and more use cases appear in a variety of sectors, holding a lot of promise for iBeacon technology.

It is highly likely some points have been missed and I invite you to send me questions, improvements, updates, use case suggestions or in fact anything relating to the topic of iBeacon and BLE.

A valuable resource for understanding BLE, referenced near the beginning of this paper, that can provide much greater technical insight is “Bluetooth® Low Energy: The Developers Handbook” by Robin Heydon.

Other valuable resources can be found at:

- https://developer.bluetooth.org/Pages/default.aspx
About Mubaloo

Mubaloo is the UK’s leading independent enterprise mobility firm according to Sourcing Line. Mubaloo focuses on enterprise app consulting, design, integration, development and deployment. Since being founded in 2009, Mubaloo has built over 180 bespoke mobile apps for leading firms covering the insurance, retail, gaming, utilities, telecoms, financial, logistics, construction, public and health sectors. Mubaloo has been recognised as App Developer of the Year 2012 & 2013 at the Appsters and holds a number of other awards for its work in creating transformative mobile tools.

MiBeacons, a Mubaloo division, was created to enable businesses to deliver contextually relevant information at the right place at the right time; through the use of beacon technology.

To find out more about building a Mobile Strategy & to further explore what’s possible for insurers, please contact:

**Mike Crooks**
Commercial Director
MiBeacons
mike.crooks@mibeacons.com

**Sarah Weller**
Managing Director (London)
Mubaloo
sarah.weller@mubaloo.com

**Bristol:**
Mubaloo
Embassy House
Queen’s Avenue
Bristol
BS8 1SB
+44 117 973 3983

**London:**
Mubaloo
2nd Floor
1 Heathcock Court
415 Strand
London
WC2R 0NT
+44 (0) 203 327 8333
Beacons:
Technical Overview

Sources

Wikipedia, Bluetooth low energy, January 3rd 2014
http://en.wikipedia.org/wiki/Bluetooth_low_energy