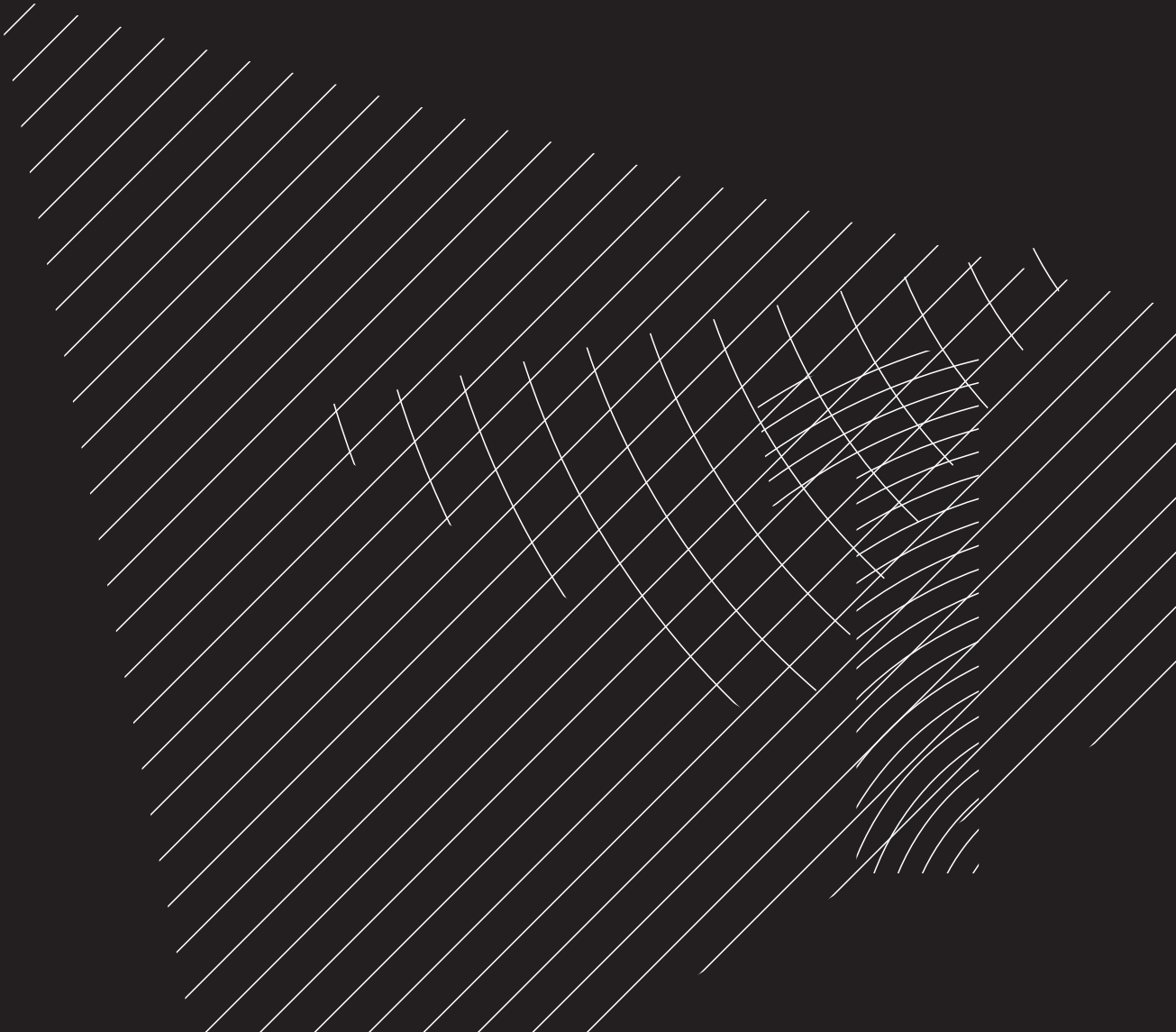


Part1. BLE 編 | 1章

# はじめに

BLE (Bluetooth Low Energy) は、  
従来の Bluetooth とはどう異なるのでしょうか？  
本章では BLE のイメージをざっくりつかんでいきます。  
(松村礼央)



「ガラケー」と呼ばれたフィーチャーフォンからスマートフォン・タブレット端末へ。この変化は、2010年代の携帯端末市場を表す代表的なトピックの1つとって過言ではありません。<sup>※1</sup>

スマートフォン・タブレット端末が従来の端末と異なる点として**アプリによるさまざまなサービスの提供と Web サービスとの連携の強化**が挙げられます。ゲームアプリで日々見知らぬ誰かと競い合う、フォトアプリで撮影した写真を見知らぬ誰かと共有し合う、そして、チャットアプリで友達と何気ない連絡事項を逐次やり取りし合う。これらは、待ち合わせの予定を駅に設置された黒板でやり取りしていた80年代では考えられませんでした。つまり、我々は意識する、しないにかかわらず、たえず誰かとつながっている、そしてそのつながりが日々強まっている世界を生きています。そして、現在。その中心にいるのは、間違いなくスマートフォン・タブレット端末です。

このインターネットと端末によってつながる世界。その次の流れとしてあるのが**ハードウェアと個人をつなげることでさまざまなサービスが実現する世界**です。このような世界を目指し、多くの研究者や技術者によっていくつもの無線技術が提案されてきました。そのうちの1つが本書の題である近距離の無線通信技術**Bluetooth Low Energy (BLE)**です。

## BLE とは何か

「BLE とは何か」を本書の読者にむけて答えるなら、冒頭で筆者 堤が述べたように、「iOS とハードウェアをワイヤレスで手軽につなげてサービスを実現する仕組み」ということに尽きます。そして、このBLEをiOSアプリから扱うフレームワーク。それが本書で取上げる、もう1つのキーワード**Core Bluetooth フレームワーク**なのです。iOS 端末/アプリにハードウェアを絡めたサービスを開発したいエンジニアにとって、「Bluetooth Low Energy (BLE)」と「Core Bluetooth フレームワーク」は、今後、避けては通れないトピックの1つとなるでしょう。

そこで本書では、BLE とそれを活用したハードウェアのハンズオンを通し、実用的な解説書となるよう執筆を行いました。本書が、iOS アプリとBLE デバイスを絡めた、読者のみなさんの新しいサービスの実現の一助となれば幸いです。

---

※1 国内のスマートフォン・タブレット端末の販売台数は2011年の2,566万台から、2016年には4,910万台までに成長すると予測されています（「スマートフォン/タブレット端末の周辺機器、アクセサリ市場と新ビジネス」、シード・プランニング,2012）。

# 1-1. BLEとは何か

Bluetooth Low Energy（以下、BLE）は「Bluetooth 4.0」で追加された新しい規格の無線技術です。「Bluetooth」という名称がついているため、従来のBluetooth 3.0（以下、クラシックBT）の進化した形態であると誤解されがちですが、**BLEとクラシックBTとはその役割や振る舞いが明確に異なります**。つまり、クラシックBTからBLEにのみ対応した機器に接続したり、逆にBLEからクラシックBTにのみ対応した機器に接続したりすることはできません。「Bluetooth 4.0」とは、これらクラシックBTとBLEとを統合した規格の名称となります。

BLEの元となった規格はNokia社の「Wibree」と呼ばれる規格です。2010年に発表されたBluetooth 4.0で採用され、現在の規格となりました。したがって、規格としては比較的若い規格といえます。

BLEは低消費電力で、通信のチャンネル数を減らし、出力を落とし、そしてパケットのサイズを軽減するなど、あらゆる面で低コストに設計されています。その結果、BLEはクラシックBTとは異なる2つの特徴を備えました。まずLow Energyという名のとおり、コイン電池で年単位の長期間の動作できる通信方式となっている点（**低電力性**）。もう1つが、そのフレームワークの下でWebサービス、アプリ、ハードウェアを連携させることで、開発者がさまざまなサービスを作ることができる点です（**サービス多様性**）。

この2点はアプリでBLEデバイスを動かす、その先を考える上で重要ですので、本書を読む上で頭の片隅にとどめておいてください。

## 1-1-1. 従来のBluetoothとBLE

では、従来のBluetooth（クラシックBT）とBLEはどのように異なるのでしょうか。ここではクラシックBTを「高級レストラン」、BLEを道の駅のような「さまざまな食材を適度な価格で売買できる市場」だとしてたとえて考えてみます。図1-1にそのイラストを示します。

1

2

3

4

5

6

7

8

9

10

11

12

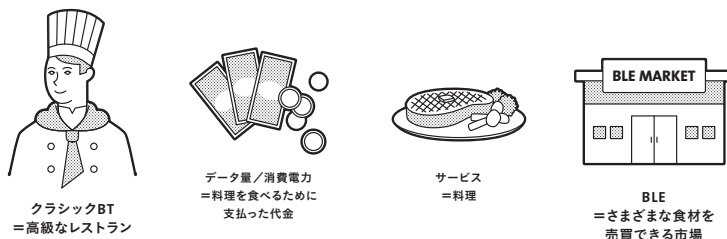


図1-1 クラシックBTとBLEをたとえて考える

図1-1のように、我々が最終的に得るサービスを「料理」だとすると、クラシックBTは「高級なレストラン」の役割を、BLEは「さまざまな食材を売買できる市場」の役割を担うものとして考えることができます。ここで、通信でやり取りされるデータは「食材」に、そしてそのデータをやり取りするために消費した消費電力は文字どおり「代金（コスト）」となります。また、データの総量は「代金の総量」に比例する、と考えてみましょう。

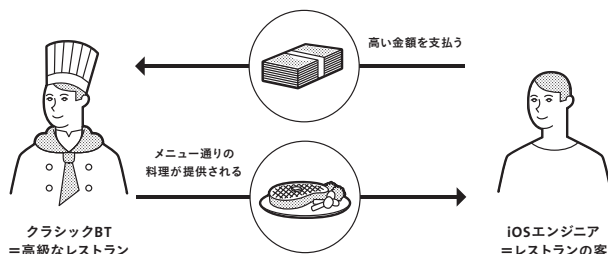


図1-2 クラシックBTの役割

**クラシックBTは調理済みの料理を提供する高級なレストランです。**代金は高額ですが、得られる料理は明らか。代金を支払うだけで望みどおりの料理を食べることができます。また、食材を持ち込んで料理を作ってもらうこともできます。しかし、高価な食材を扱うことから、持ち込む食材や調理方法については厳格な取り決めがあり、自由に食材を調理することはできません。

これは、クラシックBTが対象としているサービスが音楽、音声、静止画であることに由来します。これらのサービスは、そのデータの実時間性、連続性が非常に重要となります。音声通話がリアルタイムではなかったり、音楽が途切れ途切れになっているようなサービスでは問題です。そのためクラシックBTでは、サービスの質を中心に、通信するデータの送料や消費電力の仕様を厳格に定めています。

他方、**BLEはさまざまな食材を適度な価格で売買できる市場**です。売買される食材の代金はいずれも低額。扱うものに制限が少なく、さまざまな食材を売買できます。ここで取り引きするのは、あくまで「食材」というのがBLEのポイントです。あくまで食材なので、そのまま料

理として出せるものもあれば、組み合わせて自分なりの料理を作る必要があります。一方で、自分で料理を行うため、食材は必要なだけ、その日、そのときの料理の献立に合わせて取り引きするだけで問題はありません。

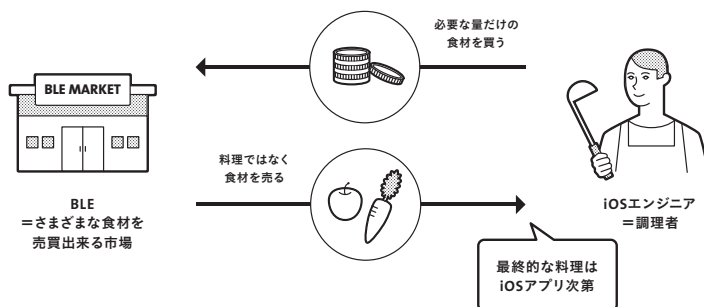


図 1-3 BLE の役割

## BLEの2つの特徴

BLEが得意とするのは、このような**必要なデータを、必要なときに、必要な分だけ**を主眼においた通信です。たとえば、体重や血圧のように一日数回計測すればよいような情報や、携帯端末からのメールやSNSの通知のように常に監視するほどでもないけれど知りたい情報などを扱うのに適しています。

このようなデータを対象とすることで、通信のやり取りをおさえ、結果として電力を抑えています。これが1つ目の特徴である**低電力性**につながっています。また当然、データ単体ではサービスとして必ずしも成立しないため、BLEではデータをアプリ側で加工することでサービスとして成立することを想定しています。つまりBLEでは、サービスの鍵の一端は読者のみなさんが担っており、そのアイデア次第でさまざまなサービスを実現することができるのです。これが2つ目の特徴である**サービス多様性**につながっています。

### 1-1-2. BLEを活用したサービス

BLEを活用したサービスで、すでに普及しつつあるのがヘルスケア・フィットネス分野です。たとえば、米NIKEの活動量計「FuelBand」が有名です。

## FuelBand

「FuelBand」は腕時計型のウェアラブルデバイスで、内蔵する三軸加速度センサから個人の活動量を計測します。従来の歩数計と異なるのは、BLEを利用してスマートフォンと連携する点です。

FuelBand自体は加速度センサから計測した情報をBLE経由でスマートフォンに通信しているだけですが、スマートフォンのアプリ側で内蔵されているGPSなどの位置情報を統合し、Webを通じて自社のサービスと連携させることで、その履歴から**どのような場所、どのような経路で、どれだけ運動したのか**などの情報を可視化してサービスとして提供しています。腕のセンサとWeb上の情報をiOSで統合することで、単なる加速度の値だった情報に付加価値を与えている点が非常に面白い製品です。



図1-4 FuelBand ([http://www.nike.com/jp/ja\\_jp/c/nikeplus-fuelband](http://www.nike.com/jp/ja_jp/c/nikeplus-fuelband))

ウェアラブルという観点でBLEを活用した製品は、FuelBand以外にも活発に開発が行われています。たとえば、「雰囲気メガネ」もその1つです。

## 雰囲気メガネ

雰囲気メガネは、スマートフォンから得られる、メール通知、ソーシャルサービス関連の通知、そしてタイマー通知などの各種通知情報を、メガネ型のデバイスにフルカラーのLEDライトと小型スピーカーからの音によって、ハンズフリーで提示します。重さは38.5gと一般的なメガネとほぼ同程度の重さながら、BLEをはじめとして、スピーカー、LED、センサ、そしてバッテリーまで備えています。ウェアラブルデバイスかつ、メガネといえ、米Googleの「Google Glass」が有名ですが、雰囲気メガネは通知情報のみにコンテンツを絞っ



図1-5 雰囲気メガネ (<http://fun-iki.com/>)

ている点がユニークです。雰囲気メガネでは、通知情報に応じた動作の制御をAPIとして開放しており、他のさまざまなサービスとの連携も想定されており、今後の展開が非常に楽しいなプロジェクトです。

ウェアラブルを志向する製品は「見ること」「確認すること」を、より自然な形態にすることで、人と機械の相互の操作をより良いものにすることが目的といえます。そのためには「電池交換が少ない」「小型である」など、**その機器を利用していることを意識させないようなシステムが望まれており**、この点でBLEの活用がされています。

この考えを個人の身体周辺のネットワークに留めず、環境自体にネットワークを埋め込むサービスへ応用したものが、iOS7で実装された「iBeacon」です。

### iBeacon

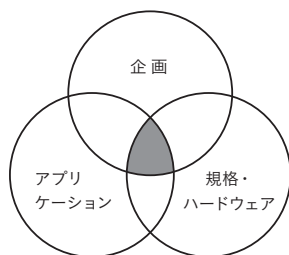
これはBLEの無線通信機能の一部である近接検出を応用しており、人が身に付けるのではなく、ランドマークとなる標識や建物に設置して利用します。文字どおりBLEを利用した「発信機（ビーコン）」として機能し、たとえば設置された屋内の位置情報とiBeaconの近接検出を応用すれば、店舗内での人の動線や購買行動を把握したうえでクーポンなどの情報を提示したり、特定の目的地に到着した際の目印としても利用することができます。

## 1-1-3. iOS エンジニアにとってのBLEの魅力（そして難しさ）

冒頭からの繰り返しになりますが、iOS エンジニアにとってのBLEの魅力は、**サービス多様性と、それを扱うフレームワーク「Core Bluetooth」がAppleから無償で公開されている**ということに尽きます。先ほどの比喻で考えると、Core BluetoothはAppleが無償で提供してくれた「キッチン」といったところでしょう。iOS エンジニアは家で自由に食材を調理できるがごとく、BLEを調理することができるのです。

しかし一方で、iOS エンジニアにとってBLEには本質的な難しさもあります。それは、**iOS エンジニア自身もハードウェアを絡めたサービスを考える側に立つ必要がある**ということです。これは家で実際に調理するときに「何を」「どうつくるか」までを含めて考えなければ、調理することはおろか、食材を調達する時点で悩んでしまうことと同様です。場合によっては、望みの食材がなければ、その食材の生産者に問い合わせるということも必要になります。

したがって、BLEを最大限に活用するためには、iOS エンジニアといえども、BLEという規格への理解とハードウェアへの理解、そして「BLEによってどのようなサービスを作るのか」という企画について、三位一体で考えることが重要となります。これがBLEの難しい点ともいえます（図1-6）。



BLEのサービスは、  
アプリケーション、サービス、規格・ハードウェアの三位一体

図1-6 BLEを活用したサービスの構成要素

## 1-1-4. Part1「BLE編」で目指すもの

そこで本書のPart1では「BLE編」と題して、iOSエンジニアがiOSアプリとBLEデバイスとの連携を体験し、BLEの規格について学ぶということを基本的な軸として、編集しました。図1-6でいうと規格・ハードウェアの領域に軸をおいています。他方、Part2「iOSプログラミング編」ではアプリケーションの領域を軸に解説しています。さらに本書の末尾では、付録として対談形式でサービスについて議論した内容を掲載しました。アプリケーション、規格・ハードウェア、企画と読者のみなさんが必要とするトピックから読み進めていただければと思います。

Part1「BLE編」では、2章で、BLEを活用したツールキットkonashiを用いて、iOSとBLEとの連携を"とりあえず"体験します。ここではプログラミングも最小限におさえ、BLEによるアプリとハードウェアの連携を体験することに焦点をあてました。ここでの内容をもとに、Part2「iOSプログラミング編」4章では、konashiをCore Bluetoothによるプログラミングによって制御します。

3章では、BLEの規格について学びます。この章については、いきなりすべてを読む必要はありません。これからBLEを学ぶ方は3章の冒頭3.1～3.3節、そしてAppleのBLEに関するガイドラインについて述べた3.11のみでも問題ありません。その後、Part2「iOSプログラミング編」以降でCore Bluetoothを学ぶにあたり、実例をベースにBLEの詳細にも踏み込むので、その際に辞書的に3章を適宜、参照していただければと思います。

それでは、さっそくiOSとBLEの世界を体験してみましょう。