

BLE デバイスを用いたリアルタイムなユーザ追跡システム

國廣 拓也[†] 藤田 聡[‡]

広島大学 工学部[†] 広島大学 大学院工学研究院[‡]

1 はじめに

BLE (Bluetooth Low Energy)は, IoT に関連する技術の中でも近年特に高い注目を集めている技術の一つである. 本稿では, 空間内に分散配置された多数の BLE デバイスを用いて, 指定された条件を満たすユーザの現在位置をリアルタイムに追跡・表示することのできるシステムを提案する. 提案システムは, Wi-Fi ネットワークで相互接続された BLE デバイス群と各ユーザが保持するスマートデバイスから構成される. BLE デバイスには, 近くを訪れたユーザの訪問時刻と滞在時間が記録され, 外部から送られてくるクエリに対してネットワークを介してリアルタイムに反応することで, 対象となるユーザの追跡を実現する.

2 BLE

BLE[1]は, 2010 年に Bluetooth 規格 4.0 で導入された低電力ワイヤレス標準であり, 4.0 以前の規格との直接の互換性はない. BLE では, 安価なハードウェアを用いて簡素で拡張性のあるデータモデルが実現されており, Android や iOS などの端末でも最近では標準で搭載されている.

BLE 通信の到達距離は, 2~5 メートルと近距離であり, その特徴を生かして, 参加者の出欠をスマートデバイスで確認するアプリケーションや, 複数の BLE デバイス間の距離を測定してクラウド上のデータベースにアップロードし分析するアプリケーションなどが開発されている. これらのアプリでは, 各 BLE デバイスに蓄えられた情報を近隣のスマートデバイスに収集して活用することが主な目的となっている.

BLE でデータを双方向に送受信するには, コネクションを使う必要がある. コネクションとは, 2 台のデバイス間での永続的で周期的なパケットデータのやり取りであり, これにはセントラルとペリフェラルの二つの役割が関係する.

データ通信の開始とタイミング管理は, セントラル(マスター)によって行われる. 提案システムでは, スマートデバイスがこの役割を担う. ペリフェラルデバイス(スレーブ)は, セントラルのタイミングにしたがい, データの送受信を定期的に行う. 提案システムでは BLE デバイスがこの役割を担う. また BLE では, ペリフェラルデバイスが発信するアドバタイズパケットをセントラルデバイスが見つかることでコネクションが確立される.

3 提案システム

提案システムは, Wi-Fi ネットワークで相互接続された BLE デバイス(以下, デバイス)群と各ユーザが保持するスマートデバイス(以下, 携帯端末)から構成される. デバイスはユーザ追跡の対象となる空間に分散配置され, デバイスの固有 ID と設置位置との対応関係(マップ)は, システムに参加するすべての携帯端末に前もって配布しておくものとする.

周辺のスキャンを行なっている携帯端末がスキャン待ち状態にあるデバイスの通信範囲に入ると, 携帯端末とデバイス間のコネクションが確立される. コネクションが確立されると, デバイスには, 携帯端末上のユーザ ID, 通信開始時刻, 通信終了時刻, 訪問回数が記録される. 通信の切断は, 携帯端末が通信範囲から離脱した時点で自動的に行われる. ここで, コネクションが確立している状態を検知することによって, そのユーザがデバイスの近くに滞在中であることが判断できることに注意しよう. なおユーザ ID は, システム利用開始時にユーザ自身で登録したニックネームで十分である.

これらの設定により, デバイス群には, a) ユーザの行動履歴を保存する分散データベースとしての機能と, b) 指定されたユーザの現在位置を検知するセンシング機能が備わることになる. 提案システムでは, これらの機能を用いてリアルタイムなユーザ追跡を次のように実現する.

User tracking system using BLE devices

[†] Takuya KUNIHIRO, Hiroshima University

[‡] Satoshi FUJITA, Hiroshima University

本研究の一部は科学研究費補助金基盤研究(B)16H02807の補助を受けています.

ユーザから見たシステムの動作は、

- 1) 検索クエリの送信,
- 2) クエリ処理,
- 3) 検索結果の受信,
- 4) 結果の UI 上への表示,
- 5) 処理終了

の指示の 5 ステップからなる。クエリの送信は、現在通信中のデバイス(以下、デバイス A)に向けて行われる(クエリの中身については後述)。受け取ったクエリは Wi-Fi ネットワークを介して他のデバイスにブロードキャストされ、クエリを受け取った各デバイスは、クエリにマッチするユーザが自分の近くに滞在中であるかを判断し、結果をデバイス A に返信する。得られた集計結果は、デバイス A からクエリを送信したユーザに返され、ユーザの端末上に表示される。またクエリにマッチする対象ユーザのリアルタイムな「追跡」を実現するため、デバイス A からのブロードキャストは、クエリ発信者の終了指示があるまで数秒間隔で定期的に繰り返される。

提案システムで想定しているクエリとしては、

- 1) ユーザ ID やデバイス ID を直接指定するタイプのクエリ,
 - 2) 通信したデバイス数の多いユーザを検索するクエリ,
 - 3) 一箇所での平均滞在時間の長いユーザを検索するクエリ,
 - 4) 総滞在時間の長いユーザを検索するクエリ,
- などがあり、最初のタイプを除いて、クエリ処理にはデバイス間の協調が必要となる。

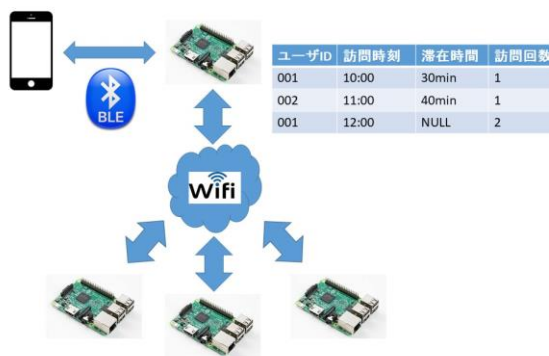


図 1: システムの概要

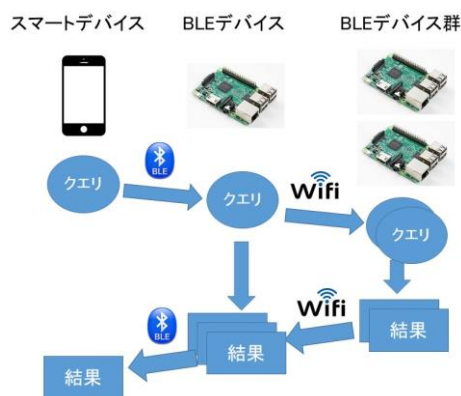


図 2: クエリ処理の流れ

Place	Arrive Time	Stay Time	Arrive Count
Place 1	10:30	25 m	1
Place 2	11:00	5 m	1
Place 1	11:30	20 m	2
Place 3	12:00	10 m	1
Place 1	12:30	25 m	3
Place 2	13:00	NULL	2

図 3: 追跡結果の表示例

4 実装システム

実装システムでは 16 台の Raspberry Pi 3 を BLE デバイス群として用い、携帯端末として ASUS 社製の ZenPad10 (OS: Android 7.0) を用いる。BLE デバイスの設置間隔は、携帯端末によるスキャン実行時に高々 1 台のデバイスが認識できるように 10m 間隔とする。16 台の BLE デバイスは Wi-Fi ネットワークに常時接続され、デバイス間の通信は WebSocket[2] を用いて行う。BLE デバイス上のプログラムは JavaScript で、携帯端末上のプログラムは Java でそれぞれ記述されている。また、デバイス上に保存される履歴データの管理は SQLite3[3] を用いて行う。

実装したシステムを使った運用実験に関しては今後の課題である。

参考文献

- [1] BluetoothSIG, <https://www.bluetooth.org/ja-jp>
- [2] WebSocket, <https://growth.pusher.com/websocket-dynamic/>
- [3] SQLite, <https://www.sqlite.org/>