

「RAD Studio 10.1 Berlinで始めるIoT アプリケーション構築」

第32回 エンバカデロ・デベロッパーキャンプ

エンバカデロ・テクノロジーズ
セールス コンサルタント 井之上 和弘



アジェンダ

- RAD Studio 10.1 BerlinでのIoTデバイスサポートについて
- RAD STUDIO で作成したアプリでIoTデバイスのデータを取得する
- IoTコンポーネントの紹介と、測定可能な値の一覧
- 位置測位アプリをBeaconfenceでコード無しに作る
- ビーコンの測距精度に影響を与える干渉源について考える



RAD STUDIO 10.1 BERLINでのIOTデバイスサポートについて

RAD Studio における IoT デバイスサポートのご紹介

- 2014/09 RAD Studio XE7
 - Bluetooth LE サポート (TBluetoothLE)



- 2015/09 RAD Studio 10 Seattle
 - BeaconFenceリリース



RAD Studio における IoT デバイスサポートのご紹介

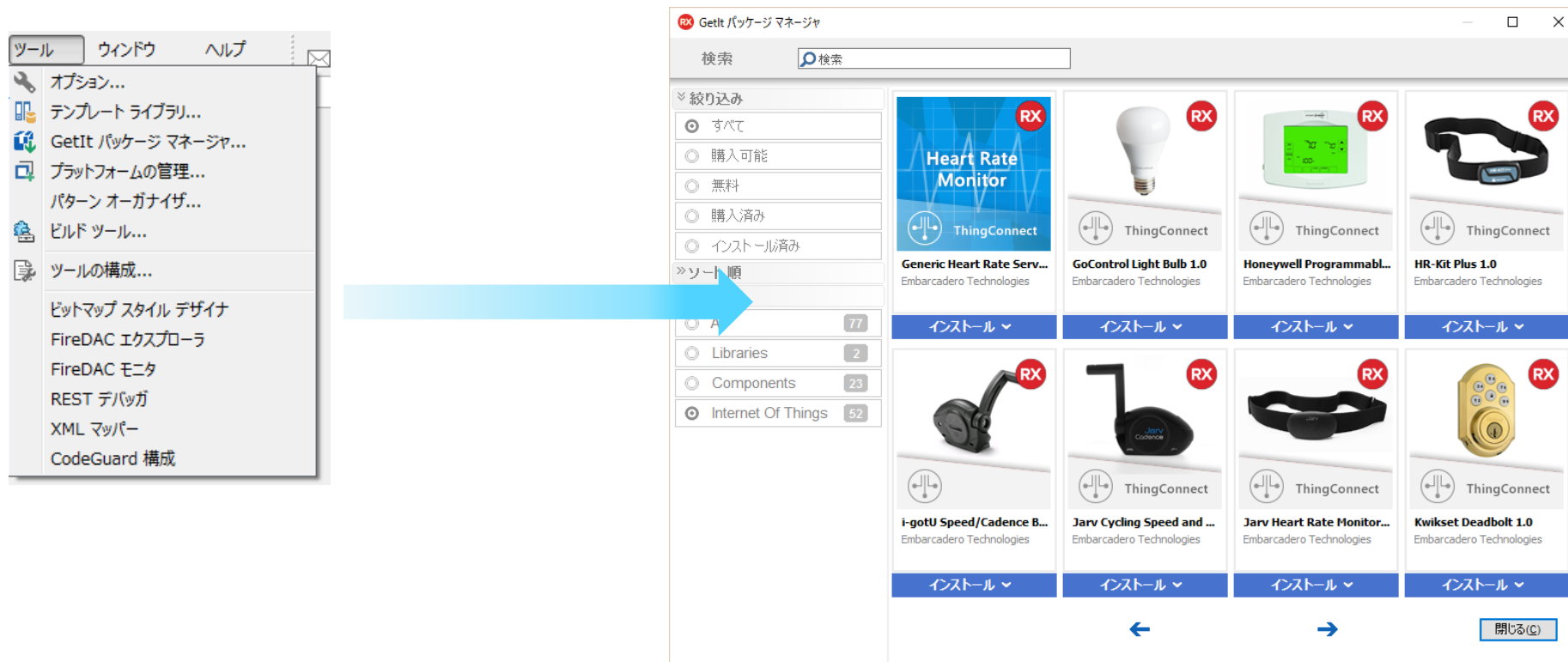
- 2016/04 RAD Studio 10.1 Berlin
 - 50を超えるIoTデバイス向けコンポーネント (Bluetooth LE、z-Wave対応) を提供開始



- Bluetooth LE
 - 基本的な無線要件は Bluetooth 3.0 と同じく 2.4GHz帯の周波数。
 - 従来のBluetoothよりも消費電力が少ないため、バッテリー駆動のウェアラブルやヘルスケア、センサーでの採用が進んでいる
- z-Wave
 - 920MHz帯の周波数。WiFiとの干渉がなく透過性も強いためにBLEよりも広い範囲をカバーできる。
 - 海外ではスマートハウスやホームセキュリティ向けの普及が進んでいる
 - 日本では2012年に認可された周波数のため、製品やサービスの展開はまだ進んでいない。

IoTコンポーネントのインストール

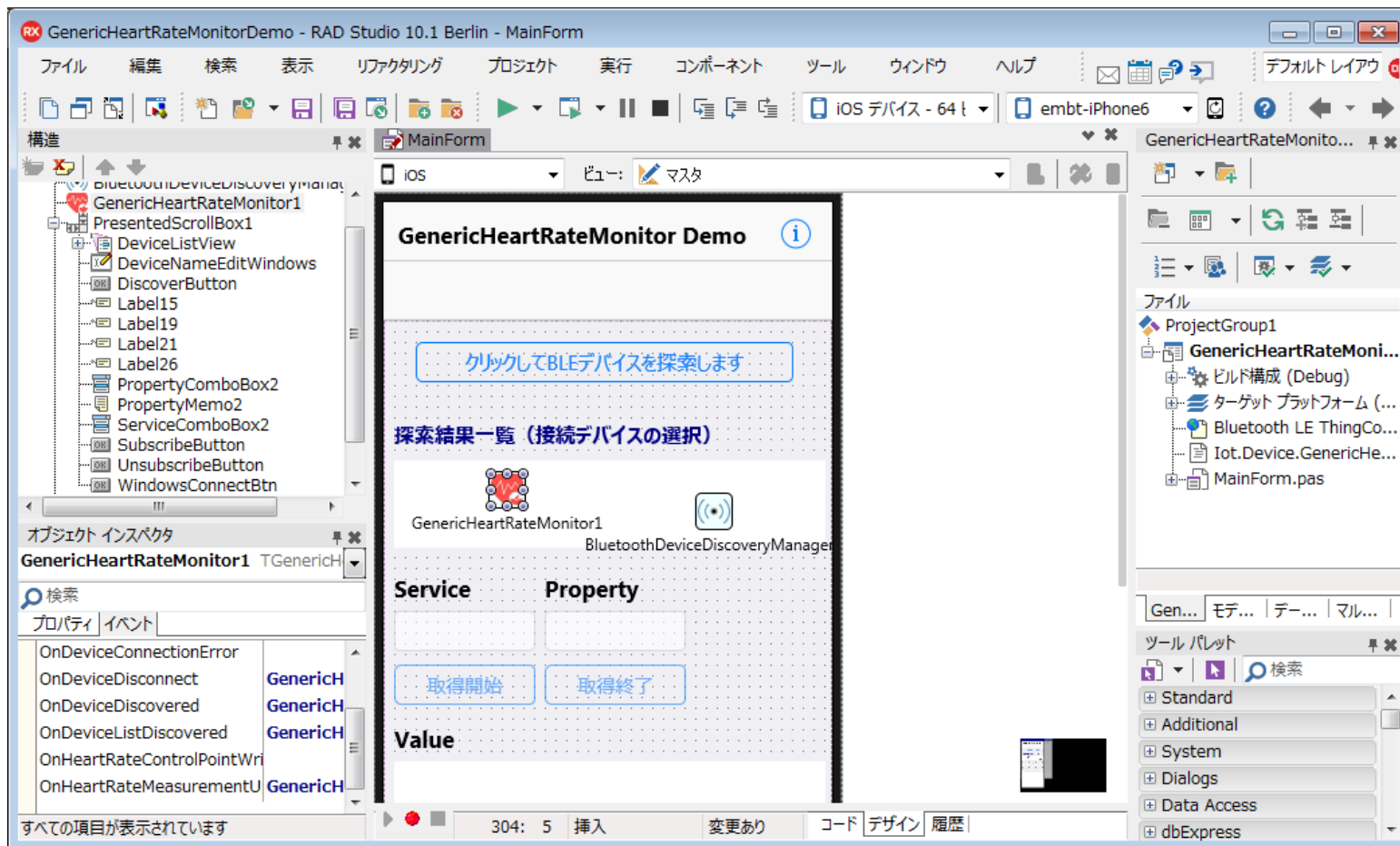
- IoTコンポーネントはGetItパッケージマネージャーからインストールする





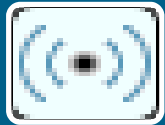
RAD STUDIO で作成したアプリで IOTデバイスのデータを取得する

心拍計の値を取得するアプリの例



使用するコンポーネント

- BLEのIoTセンサーから値を読む場合は2種類のコンポーネントを用いる



BluetoothDeviceDiscoveryManager

- BluetoothLE デバイスの検出を行うコンポーネント



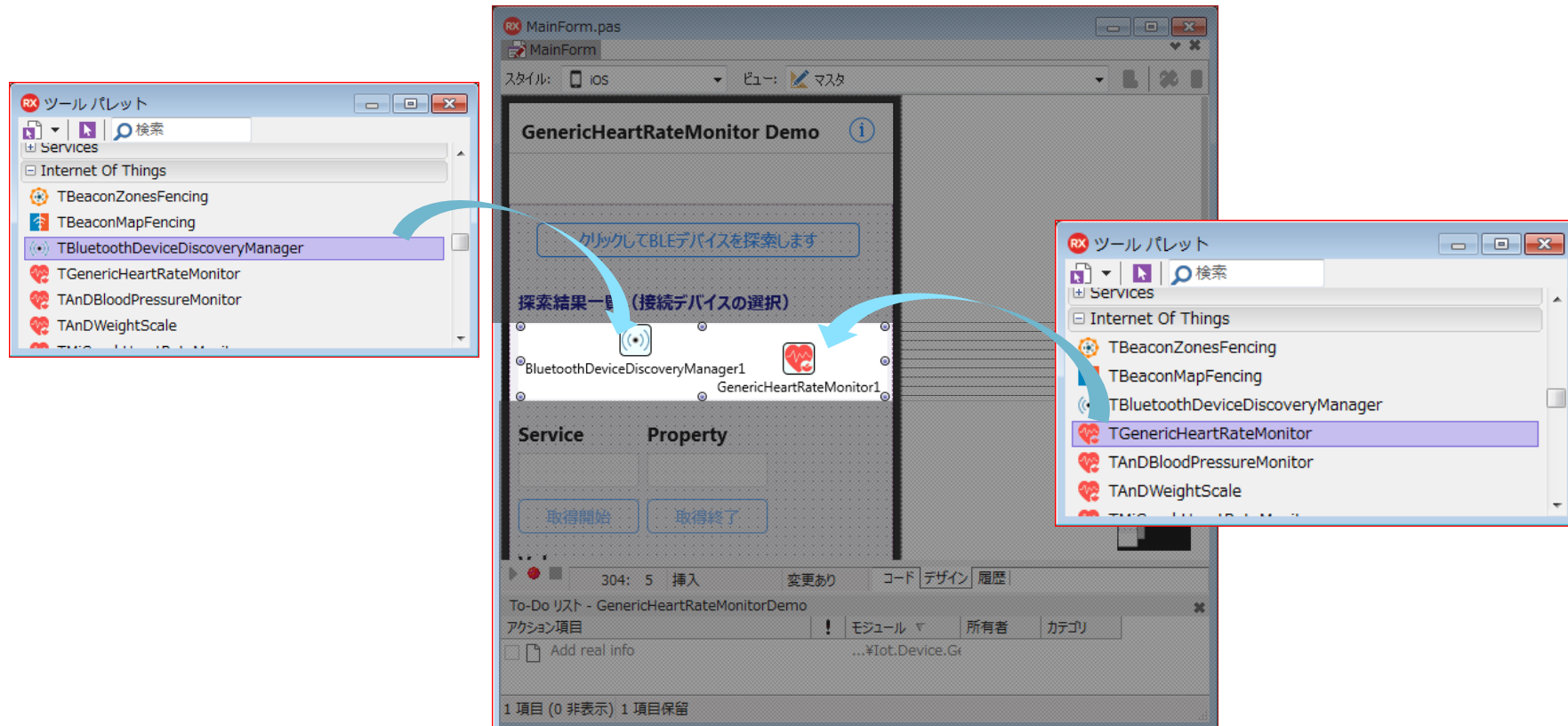
デバイス毎の専用コンポーネント

(ここでは GenericHeartRateMonitor を使用)

- センサーが提供する機能（サービス）に適したコンポーネントを選択する

デザインフォームにコンポーネントを配置する

- BluetoothDeviceDiscoveryManagerとGenericHeartRateMonitorを配置する。

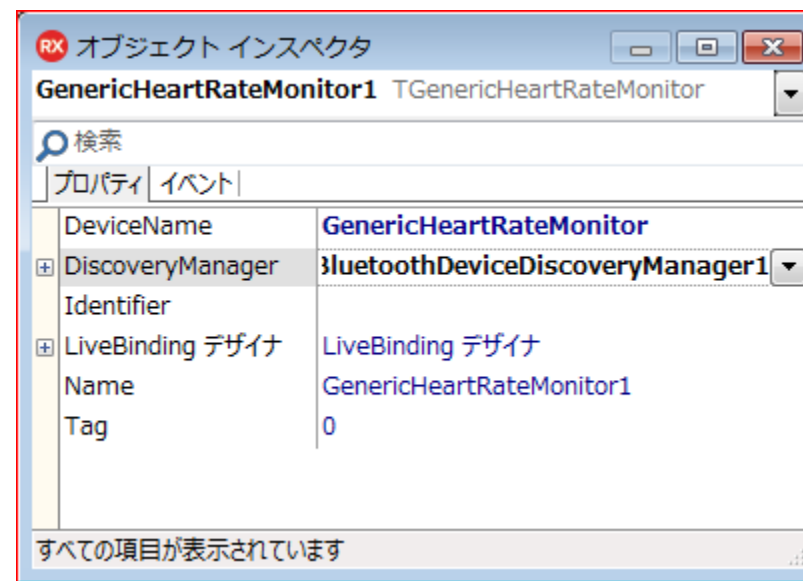


コンポーネントのプロパティを設定する

- BluetoothDeviceDiscoveryManager の"DiscoveryMethod"を目的や用途に合わせて設定する。



- GenericHeartRateMonitor の"DiscoveryManager"に "BluetoothDeviceDiscoveryManager" を設定する。



デバイスに接続するためのコードを書く

- デバイスに接続するには、コードからDiscoverDevices を実行する。

Delphi

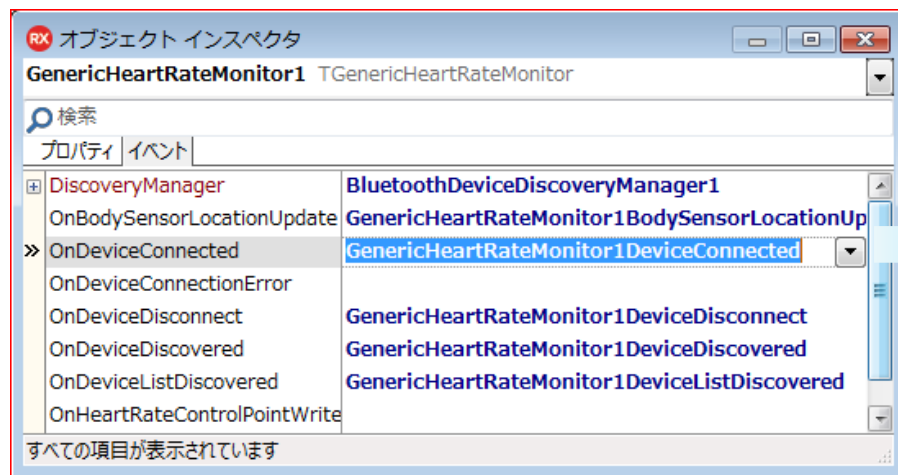
```
BluetoothDeviceDiscoveryManager1.DiscoverDevices;
```

C++

```
BluetoothDeviceDiscoveryManager1->DiscoverDevices;
```

GenericHeartRateMonitorのイベントに紐づくコードを実装する

- デバイスへの接続が完了すると OnDeviceConnected のイベントが発生するので、必要な処理を実装する。



```
procedure TForm1.GenericHeartRateMonitor1DeviceConnected;  
begin  
    Status.Text := 'Connected';  
    if ServiceComboBox.Items.Count > 0 then  
    begin  
        ServiceComboBox.Enabled := True;  
        PropertyComboBox.Enabled := True;  
    end;  
    DisconnectButton.Visible := True;  
end;
```

- その他のイベントについても必要な処理を実装する。

IoTアプリケーションの実装に関するdocwiki上の文書やサンプルコード

- Working with ThingConnect Devices
 - http://docwiki.embarcadero.com/IoT/en/Working_with_ThingConnect_Devices
- Tutorial: Building a ThingConnect IoT Application
 - http://docwiki.embarcadero.com/IoT/en/Tutorial:_Building_a_ThingConnect_IoT_Application
- Sample Project
 - C:\Users\Public\Documents\Embarcadero\Studio\18.0\Samples\Internet of Things\Object Pascal\Thing Connect
 - C:\Users\Public\Documents\Embarcadero\Studio\18.0\Samples\Internet of Things\CPP\Thing Connect

まとめ

- BLEデバイスのIoTコンポーネントを用いる場合は2つのコンポーネントを使用します。
 - BluetoothDeviceDiscoveryManager
 - 使用するデバイスごとのコンポーネント
- アプリケーションの実装は通常の Delphi/C++ アプリケーションの開発と同じように行えます。



IOTコンポーネントの紹介と、測定 可能な値の一覧

IoTコンポーネントの紹介(BLE)

■ 10.1 Berlin に対応のデバイスの種類と対応数

種類	対応デバイス数	汎用コンポーネントの有無	使用サービス
 心拍計	8	あり	Heart Rate
 血圧計	3	あり	Blood Pressure
 体温計	1	あり	Health Thermometer
 体重計	5	なし	Custom Service
 スピード&ケイデンス(自転車向け)	10	なし※	Cycling Speed and Cadence
 スピード&ケイデンス(ランニング向け)	1	なし※	Running Speed and Cadence
 活動量計	1	なし	Custom Service
 ジェスチャートラッカー	1	なし	Custom Service
 環境センサー	3	なし	Custom Service
 BeaconFence	1		

心拍計



Service	Characteristic	Property	Readable	Subscription	Write	Coverage	Device							
							Generic Heart Rate Monitor	Janv BT Run Smart Heart Monitor	Polar H7 Heart Rate Sensor	Adidas Micoach	Wahoo TICKR	Zephyr Heart Rate Monitor	Runtastic Heart Rate Combo Monitor	HR-Kit Plus
Device Information	System ID	SystemID	6	0	0	75.0%			R__	R__	R__	R__	R__	R__
	Model Number String	ModelNumber	5	0	0	62.5%			R__		R__	R__	R__	R__
	Serial Number String	SerialNumber	7	0	0	87.5%		R__	R__	R__	R__	R__	R__	R__
	Firmware Revision String	FirmwareRevision	7	0	0	87.5%		R__	R__	R__	R__	R__	R__	R__
	Hardware Revision String	HardwareRevision	6	0	0	75.0%		R__	R__		R__	R__	R__	R__
	Software Revision String	SoftwareRevision	5	0	0	62.5%			R__		R__	R__	R__	R__
	Manufacturer Name String	ManufacturerName	7	0	0	87.5%		R__	R__	R__	R__	R__	R__	R__
	IEEE 11073-20601 Regulatory Certification Data List	IEEERegulatory	1	0	0	12.5%						R__		
	PnP ID	PnPID	2	0	0	25.0%		R__				R__		
Battery Service	Battery Level	BatteryLevel	7	5	0	87.5%		RS_	R__	RS_	R__	RS_	RS_	RS_
Heart Rate	Heart Rate Measurement	HeartRateMeasurement (*)	0	8	0	100.0%	_S_	_S_	_S_	_S_	_S_	_S_	_S_	_S_
	Body Sensor Location	BodySensorLocation	8	0	0	100.0%	R__	R__	R__	R__	R__	R__	R__	R__
	Heart Rate Control Point	HeartRateControlPoint (**)	0	0	1	12.5%	__W							

血圧計



Service	Characteristic/Description	Property	Readable	Subscription	Write	Coverage	device		
							Generic Blood Pressure Monitor	A&D Blood Pressure Monitor	Pyle Blood Pressure Monitor
Device Information	System ID	SystemID	3	0	0	33.3%		R__	
	Model Number String	ModelNumber	1	0	0	33.3%		R__	
	Serial Number String	SerialNumber	1	0	0	33.3%		R__	
	Firmware Revision String	FirmwareRevision	1	0	0	33.3%		R__	
	Hardware Revision String	HardwareRevision	1	0	0	33.3%		R__	
	Software Revision String	SoftwareRevision	1	0	0	33.3%		R__	
	Manufacturer Name String	ManufacturerName	1	0	0	33.3%		R__	
	IEEE 11073-20601 Regulatory Certification Data List	IEEERegulatory	1	0	0	33.3%		R__	
Battery Service	Battery Level	BatteryLevel	2	0	0	66.7%		R__	R__
Blood Pressure	Blood Pressure Measurement	BloodPressureMeasurement(*)	0	3	0	100.0%	_S_	_S_	_S_
	Intermediate Cuff Pressure	IntermediateCuffPressure(*)	0	1	0	33.3%	_S_		
	Blood Pressure Feature	BloodPressureFeature	2	0	0	66.7%	R__	R__	
	Date Time	DateTime	1	0	1	33.3%		R_W	
Custom Service 1	Returns theTANDBloodPressureCommandSerializeddata for the custom services.	AnDCustomCommandStatus	1	0	1	33.3%		R_W	

体温計



							device
<u>Service</u>	<u>Characteristic</u>	Property	Readable	Subscription	Write	Coverage	Pyle Health Thermometer
<u>Battery Service</u>	<u>Battery Level</u>	BatteryLevel	1	0	0	100.0%	R_
<u>Health Thermometer</u>	<u>Temperature Measurement</u>	TemperatureMeasurement (*)	0	1	0	100.0%	_S_
	<u>Temperature Type</u>	TemperatureType	1	1	0	100.0%	RS_
	<u>Intermediate Temperature</u>	IntermediateTemperature (*)	0	1	0	100.0%	_S_
	<u>Measurement Interval</u>	MeasurementInterval	1	0	0	100.0%	R_

体重計



<u>Service</u>	<u>Characteristic</u>	Property	device								
			Readable	Subscription	Write	Coverage	Wahoo Weight Scale Monitor	A&D Weight Scale	Pyle Health Weight Scale	Taylor Smart Scale	Weight Gurus Scale
<u>Device Information</u>	<u>System ID</u>	SystemID	5	0	0	100.0%	R__	R__	R__	R__	R__
	<u>Model Number String</u>	ModelNumber	4	0	0	80.0%		R__	R__	R__	R__
	<u>Serial Number String</u>	SerialNumber	5	0	0	100.0%	R__	R__	R__	R__	R__
	<u>Firmware Revision String</u>	FirmwareRevision	5	0	0	100.0%	R__	R__	R__	R__	R__
	<u>Hardware Revision String</u>	HardwareRevision	5	0	0	100.0%	R__	R__	R__	R__	R__
	<u>Software Revision String</u>	SoftwareRevision	4	0	0	80.0%		R__	R__	R__	R__
	<u>Manufacturer Name String</u>	ManufacturerName	5	0	0	100.0%	R__	R__	R__	R__	R__
	<u>IEEE 11073-20601 Regulatory Certification Data List</u>	IEEERegulatory	3	0	0	60.0%		R__	R__	R__	
<u>PnP ID</u>	PnPID	2	0	0	40.0%			R__	R__		
<u>Battery Service</u>	<u>Battery Level</u>	BatteryLevel	3	0	0	60.0%	R__	R__	R__		

※体重計向けの汎用コンポーネントはない

スピード&ケイデンス(自転車向け)



Service	Characteristic	Property	Readble	Subscription	Write	Coverage	device									
							Wahoo CSC Sensor	Garmin Cycling Speed and Cadence Bike Sensor	Leotlu Speed/Cadence Bike Sensor	Topack Speed/Cadence Sensor	ARMOR-X Cycling Speed/Cadence Sensor	PowerTap Speed and Cadence Sensor	Runtastic Speed/Cadence Bike Sensor	Polar Speed Sensor	Polar Cadence Sensor	Scosche Rhythm Speed/Cadence Sensor
Device Information	Firmware Revision String	FirmwareRevision	7	0	0	90.0%		R__	R__	R__	R__	R__	R__	R__	R__	R__
	Hardware Revision String	HardwareRevision	5	0	0	60.0%		R__	R__	R__	R__			R__	R__	
	Manufacturer Name String	ManufacturerName	7	0	0	90.0%		R__	R__	R__	R__	R__	R__	R__	R__	R__
	Model Number String	ModelNumber	3	0	0	40.0%				R__	R__			R__	R__	
	Serial Number String	SerialNumber	7	0	0	90.0%		R__	R__	R__	R__	R__	R__	R__	R__	R__
	Software Revision String	SoftwareRevision	5	0	0	60.0%		R__	R__	R__	R__			R__	R__	
	System ID	SystemID	5	0	0	70.0%				R__	R__	R__	R__	R__	R__	R__
PnP ID	PnPID	3	0	0	30.0%		R__	R__		R__						
Battery Service	Battery Level	BatteryLevel	6	5	0	70.0%	R__	RS_	RS_	RS_	RS_			RS_	RS_	
Cycling Speed and Cadence	CSC Feature	CSCFeature	8	0	0	100.0%	R__	R__	R__	R__	R__	R__	R__	R__	R__	R__
	CSC Measurement	CumulativeWheelRevolutions(*)	0	8	0	90.0%	_S_	_S_	_S_	_S_	_S_	_S_	_S_	_S_		_S_
		CumulativeCrankRevolutions(*)	0	7	0	90.0%	_S_	_S_	_S_	_S_	_S_	_S_	_S_		_S_	_S_
		CSCMeasurement	0	0	0	10.0%										R__
	SC Control Point	SCControlPoint(**)	0	7	7	80.0%	_SW	_SW	_SW	_SW	_SW	_SW		_SW		R__
	Sensor Location	SensorLocation	6	0	0	70.0%	R__	R__	R__	R__	R__	R__				R__
	Sensor Location	CSCSensorLocation	1	0	0	10.0%							R__			

スピード&ケイデンス(ランニング向け)



Service	Characteristic	Property	Readable	Subscription	Write	Coverage	Polar Stride Sensor
							device
Device Information	Firmware Revision String	FirmwareRevision	1	0	0	100.0%	R_
	Hardware Revision String	HardwareRevision	1	0	0	100.0%	R_
	Manufacturer Name String	ManufacturerName	1	0	0	100.0%	R_
	Model Number String	ModelNumber	1	0	0	100.0%	R_
	Serial Number String	SerialNumber	1	0	0	100.0%	R_
	Software Revision String	SoftwareRevision	1	0	0	100.0%	R_
	System ID	SystemID	1	0	0	100.0%	R_
	PnP ID	PnPID	0	0	0	0.0%	
Battery Service	Battery Level	BatteryLevel	1	1	0	100.0%	RS_
Running Speed and Cadence	RSC Feature	RunningSpeedCadence Feature	1	0	0	100.0%	R_
	CSC Measurement	InstantaneousCadence(*)	0	1	0	100.0%	_S_
		InstantaneousSpeed(*)	0	1	0	100.0%	_S_
	SC Control Point	SCControlPoint(**)	0	1	1	100.0%	_SW

活動量計



<u>Service</u>	<u>Characteristic</u>	Property	Readable	Subscription	Write	Coverage	device
							Lifetrak Move Watch
<u>Device Information</u>	<u>System ID</u>	SystemID	1	0	0	100.0%	R__
	<u>Model Number String</u>	ModelNumber	1	0	0	100.0%	R__
	<u>Serial Number String</u>	SerialNumber	1	0	0	100.0%	R__
	<u>Firmware Revision String</u>	FirmwareRevision	1	0	0	100.0%	R__
	<u>Hardware Revision String</u>	HardwareRevision	1	0	0	100.0%	R__
	<u>Software Revision String</u>	SoftwareRevision	1	0	0	100.0%	R__
	<u>Manufacturer Name String</u>	ManufacturerName	1	0	0	100.0%	R__
	<u>IEEE 11073-20601 Regulatory Certification Data List</u>	IEEECertificationData	1	0	0	100.0%	R__
	<u>PnP ID</u>	PnPID	1	0	0	100.0%	R__
<u>Battery Service</u>	<u>Battery Level</u>	BatteryLevel	1	1	0	100.0%	RS_

※活動量計向けの汎用コンポーネントはない

ジェスチャートラッカー



							device
<u>Service</u>	<u>Characteristic</u>	Property	Readable	Subscription	Write	Coverage	Myo Gesture Control Armband
<u>Device Information</u>	<u>Manufacturer Name String</u>	ManufacturerName	1	0	0	100.0%	R_
<u>Battery Service</u>	<u>Battery Level</u>	BatteryLevel	1	1	0	100.0%	RS_

※ジェスチャートラッカー向けの汎用コンポーネントはない

環境センサー



Service	Characteristic	Property	Readable	Subscription	Write	Coverage	device		
							Texas Instruments Sensor Tag	Texas Instruments Gas Sensor Platform	Tempo Environment Monitor
Device Information	System ID	SystemID	1	0	0	33.3%	R__		
	Model Number String	ModelNumber	2	0	0	66.7%	R__		R__
	Serial Number String	SerialNumber	1	0	0	33.3%	R__		
	Firmware Revision String	FirmwareRevision	2	0	0	66.7%	R__		R__
	Hardware Revision String	HardwareRevision	2	0	0	66.7%	R__		R__
	Software Revision String	SoftwareRevision	2	0	0	66.7%	R__		R__
	Manufacturer Name String	ManufacturerName	2	0	0	66.7%	R__		R__
	IEEE 11073-20601 Regulatory Certification Data List	IEEERegulatory	1	0	0	33.3%	R__		
PnP ID	PnPID	1	0	0	33.3%	R__			
Battery Service	Battery Level	BatteryLevel	2	0	0	66.7%		R__	R__
Immediate Alert	Alert Level	ImmediateAlert	0	0	1	33.3%			__W

※環境センサー向けの汎用コンポーネントはない

IoTコンポーネントの紹介(z-Wave)

- z-Wave向けのコンポーネントは以下の種類があります。

種類	対応デバイス数
Alarm	3
Doorlock	3
Switch	2
Outlet	3
Light	2
Sensor	5

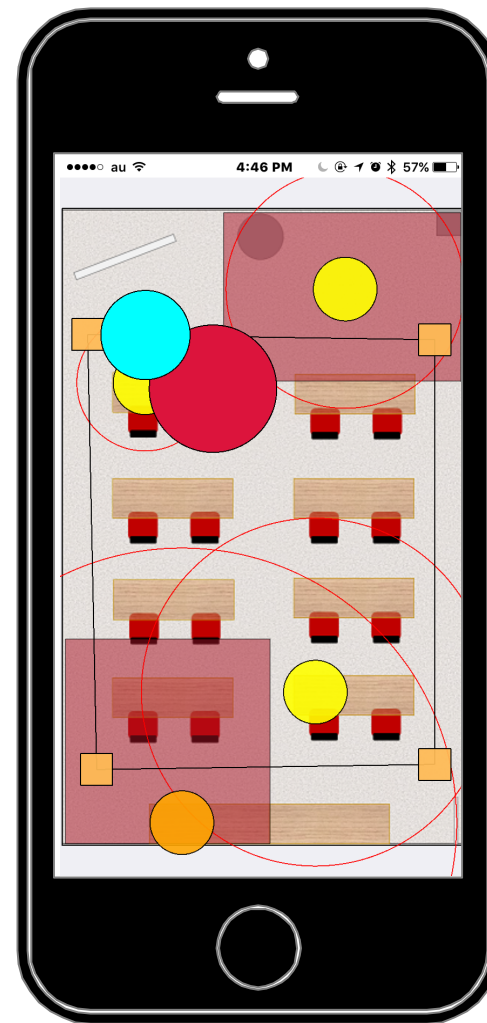
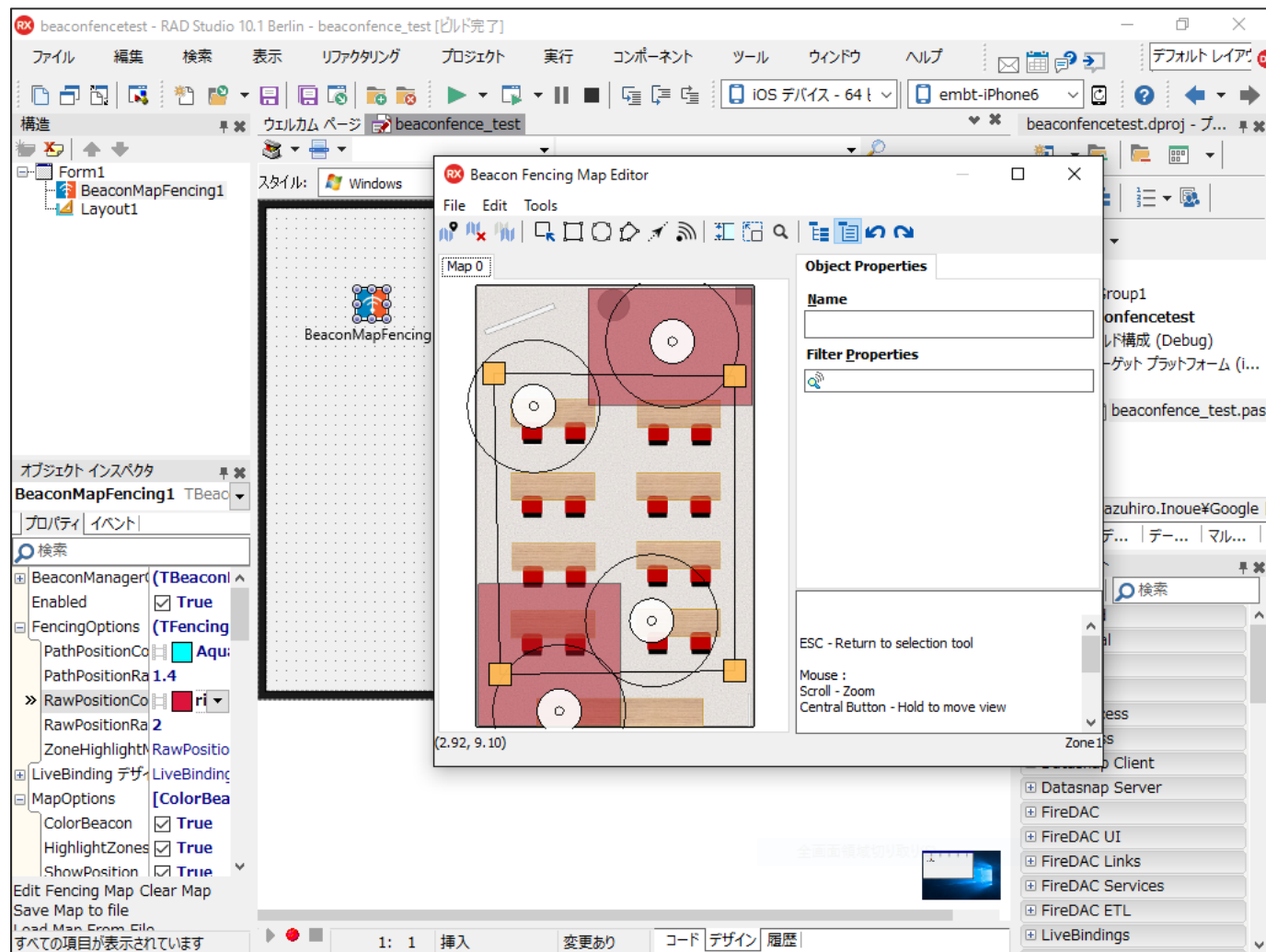
See also

- 対応デバイス一覧
 - http://docwiki.embarcadero.com/IoT/en/ThingConnect_Devices



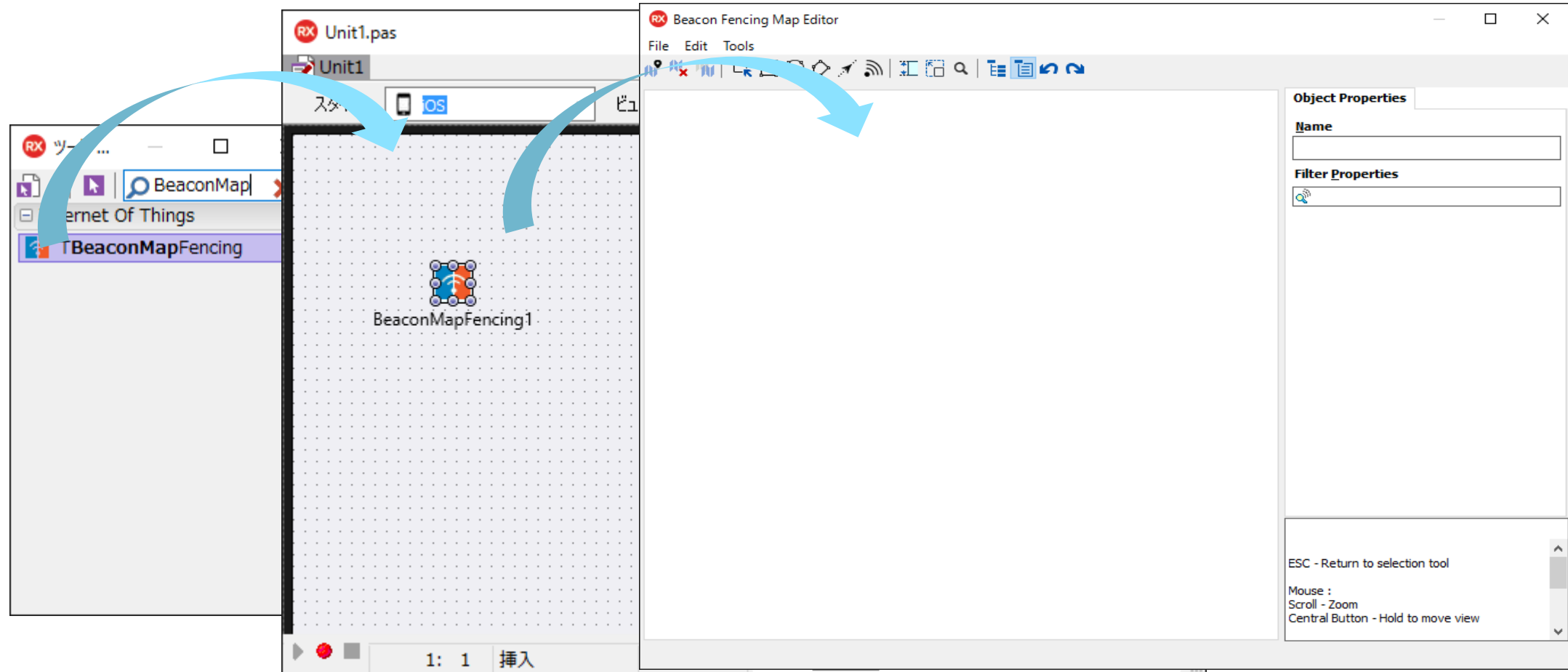
位置測位アプリをBEACONFENCE でコード無しに作る

Beaconfenceで測位アプリを作る。



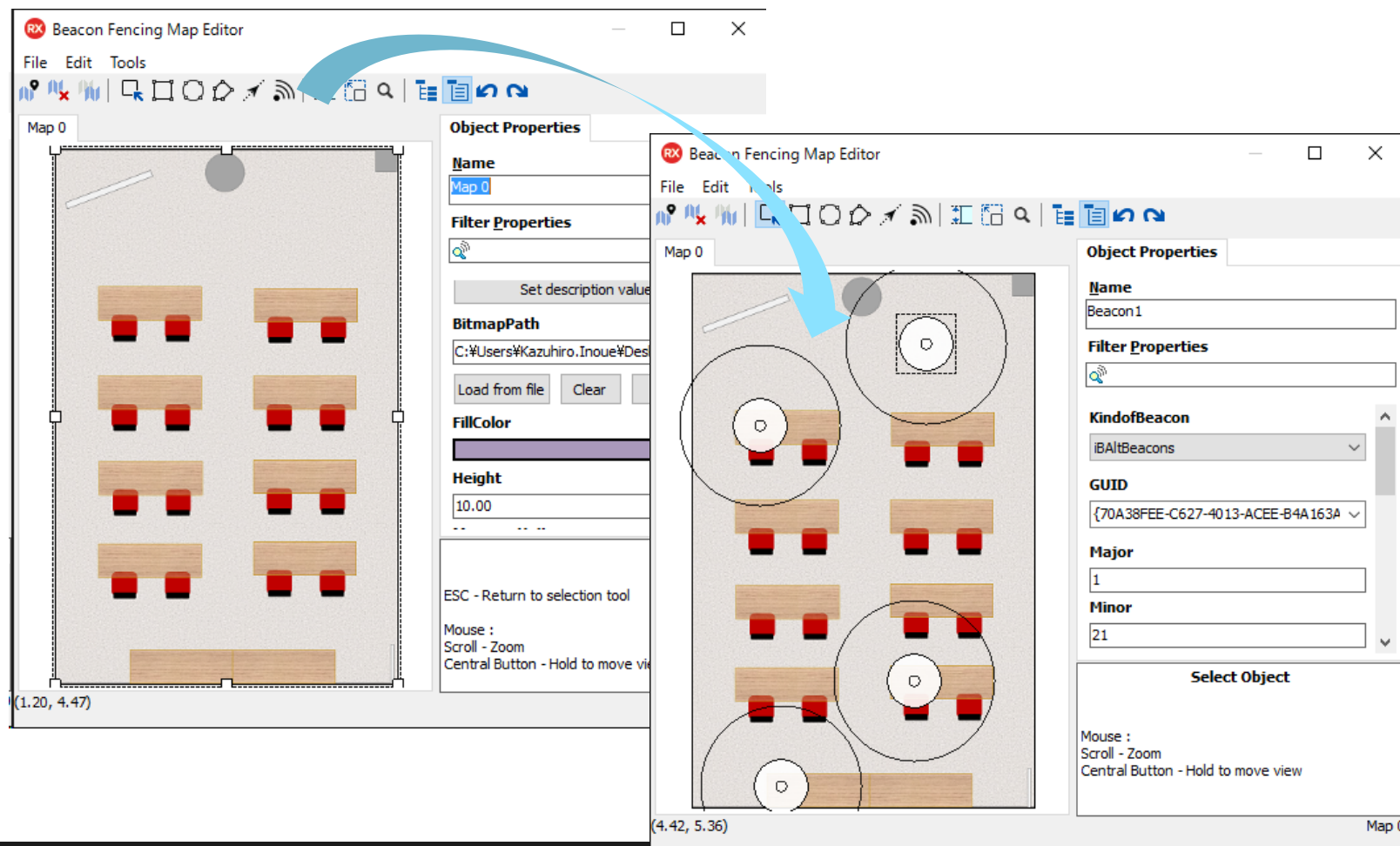
Beaconfenceで測位アプリを作る。

- TBeaconMapFencingを配置する



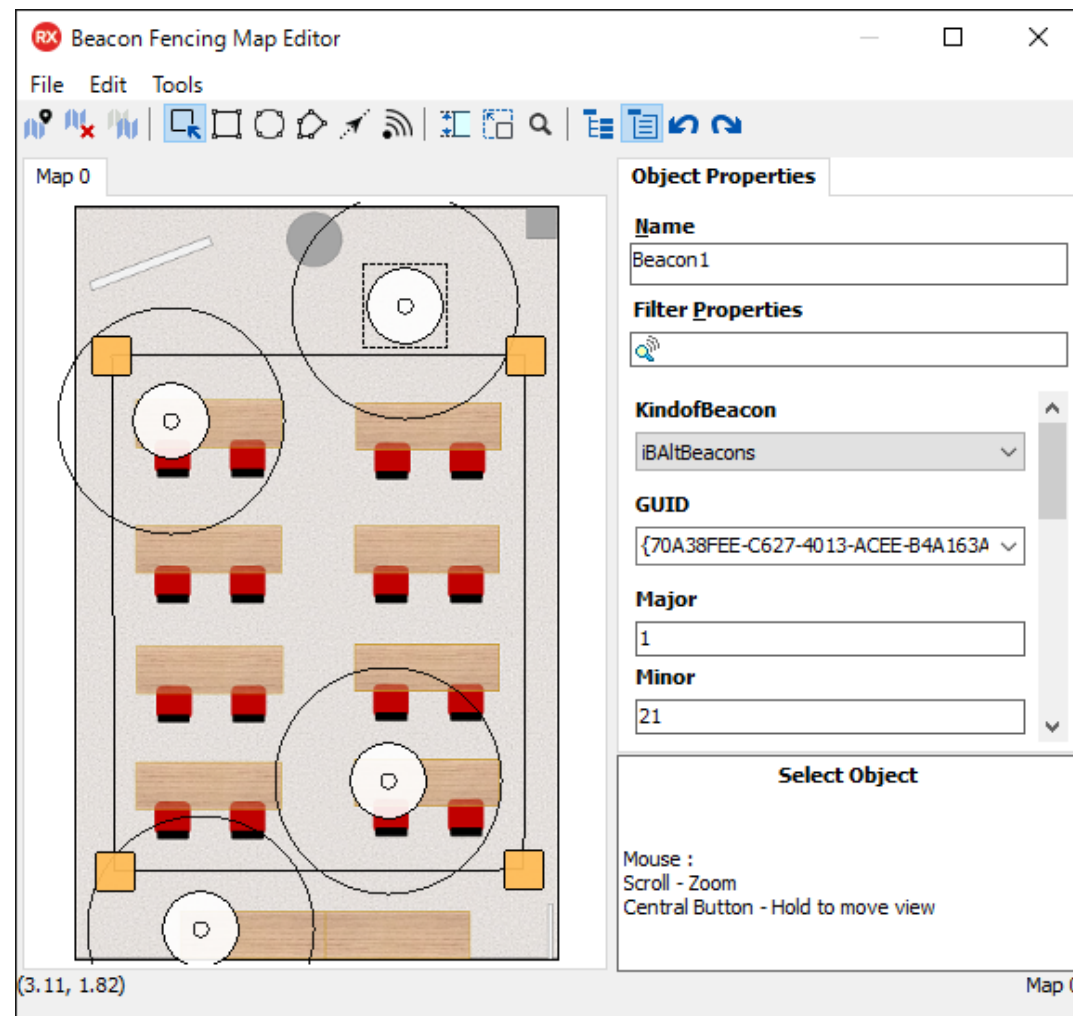
Beaconfenceで測位アプリを作る。

- フロアマップを定義し、ビーコンを配置する。



Beaconfenceで測位アプリを作る。

- 測位精度を高めるために、人の動線を「パス」で定義する。
 - 「パス」は、カーナビにおける道路に相当する。
 - 利用者は「パス」の上だけを移動するものと見なす。計算から求めた位置をパス上に補正する。
 - ビーコンの測距、測位精度はさほど高くないが、パスを定義することで測位位置の補正が行える。



BeaconFenceを用いたアプリケーション実装に関するdocwiki上の文書

- BeaconFence
 - <http://docwiki.embarcadero.com/IoT/en/BeaconFence>
- Using BeaconFence
 - <http://docwiki.embarcadero.com/IoT/en/Tutorial: Building a ThingConnect IoT Application>
- BeaconFence Map Editor
 - http://docwiki.embarcadero.com/IoT/en/BeaconFence_Map_Editor

まとめ

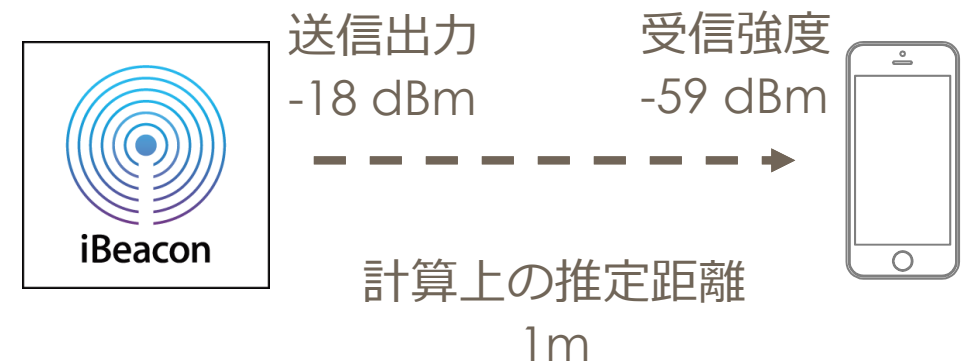
- BeaconFenceでは位置測位だけを行うアプリケーションならば、コードをほとんど書かずに実装できる！
- BeaconFenceは、固定ビーコンを使用した位置測位が行える。
- 測位の精度を上げるには「パス」を設定する。



ビーコンの測距精度に影響を与える 干渉源について考える

IoTセンサーとビーコンの要件の違い

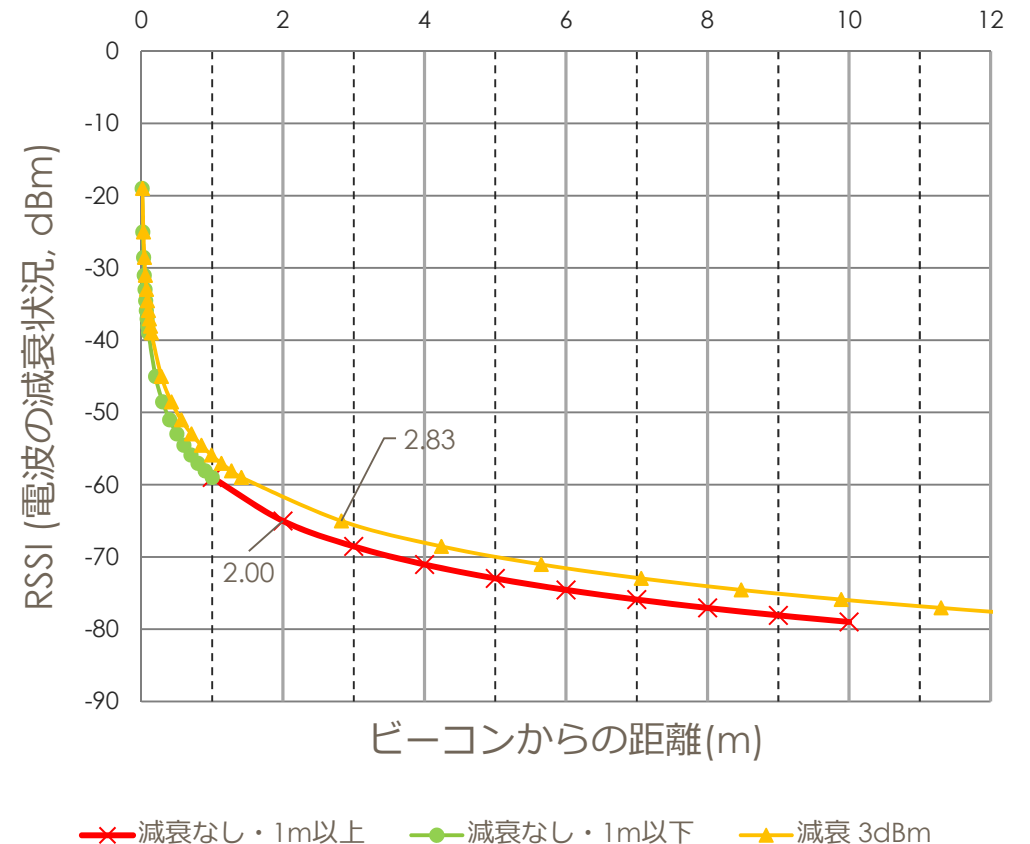
- IoTセンサーは、計測値が取得できればよい。
- 電波強度が少々下がっても値が取得できるならば特に差し支えない。
- ビーコンでは「距離」や「近接度」を測定したい。
- 「距離」は送信出力と受信強度から算出するため、受信強度の数値が重要。
- 受信強度が不安定な条件では、距離計算のブレ幅も大きくなる。送信強度が不安定なのは問題外。



距離による電波強度の減衰について考える

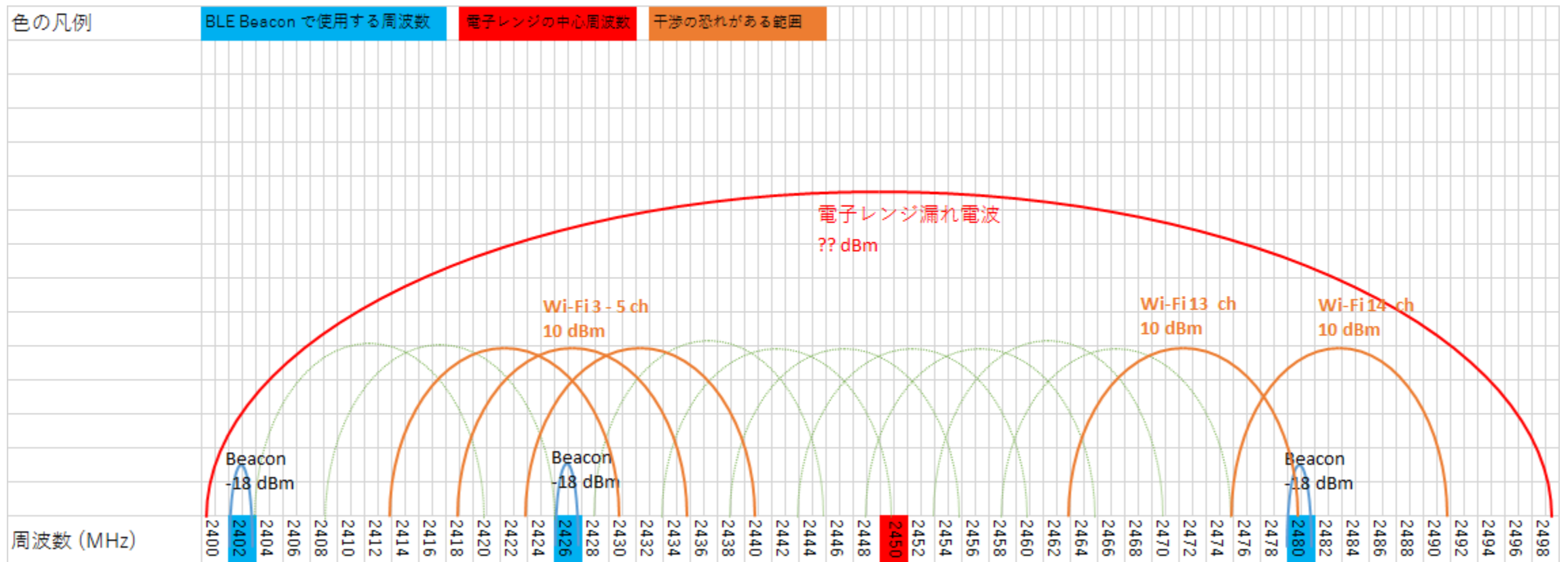
- 電波は距離の2乗に反比例して弱くなる。
- ビーコンとの距離が1メートルより遠いと距離検出の精度が落ちる。
 - RSSIの値は1メートルを超えると数値の下がり方が小さくなる。つまり**数値の少しの差が距離計算に与える影響が大きい。**
- **3dBm 程度の減衰でもビーコンから2m程度離れた位置では測距が1m近くズれる。**（減衰によって距離の計算に40%程度のズレが出るため）

ビーコンの出力強度と距離による減衰の関係



BLE Beaconの電波周波数と主な干渉源

- BLE Beaconの電波は電子レンジの漏れ電波やWi-Fiの一部チャンネルの電波の干渉を受ける。

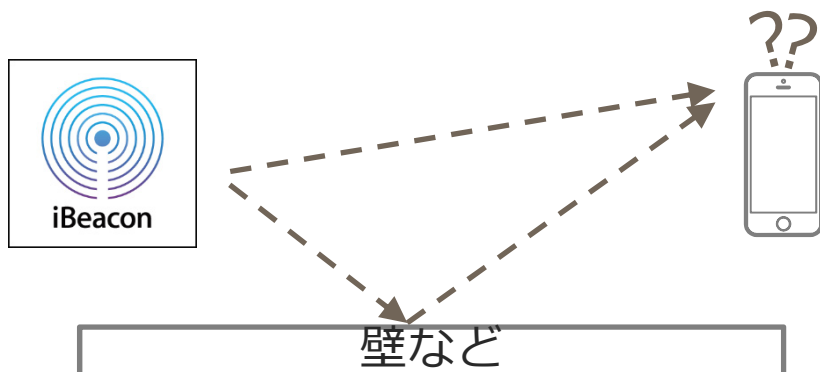


干渉する恐れのあるデバイス

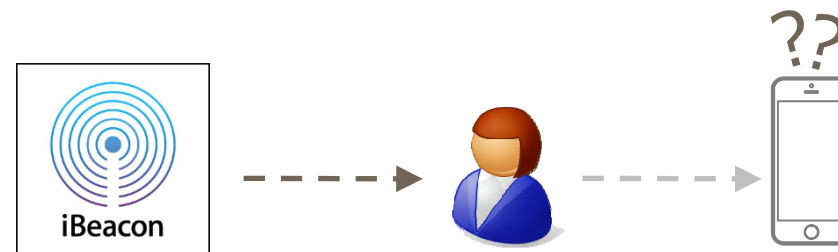
- 電子レンジ
 - 2.45GHzを中心周波数とし、2.4GHz-2.5GHzの範囲で高出力の電波を放射する。
 - Bluetoothデバイスに比べて出力が桁違いに大きい(電子レンジ = 400W以上、BLEデバイス = 0.01W)。電子レンジの電波は筐体によって遮蔽されているが、それでも発生する漏れ電波のインパクトが大きい。
- WiFi
 - 2.4GHz 802.11b/g/n と Beacon を併用する場合は、3-5, 13-14ch の使用を極力避けることを検討してみる。このチャンネルは BLE デバイスと干渉する恐れがある。
- デジタルコードレス電話
 - 一部のデジタルコードレス電話は2.4GHz帯を使用するため、BLEデバイスとの干渉の恐れがある。

その他の干渉源

- マルチパス
 - 直接届く電波と反射波の組み合わせによって、電波強度が本来よりも強く出る、あるいは弱く出る現象。
 - アナログテレビ放送で言うところのゴースト



- 水を多く含む物体（生体含む）
 - 2.4GHz帯の電波は水に吸収されやすいため、水を多く含む物体によって電波が減衰する可能性がある。



まとめ

- IoTセンサーとビーコンはBluetooth LEという点では共通だが、取り扱うデータに求められる要件は異なる。
- ビーコンの設置にはさまざまなノウハウがあるが、技術要件から事前に想定可能な問題点は考慮の上でビーコンの配置を設計することが望ましい。

THANKS!

www.embarcadero.com/jp

第32回 エンバカデロ・デベロッパーキャンプ