

Cocos2d-x x iBeacon ?

Bluetoothを使ったゲームを作ろう

アクセルマーク株式会社

Tomoo Kaku

自己紹介

- Tomoo Kaku
- アクセルマーク株式会社 シニアエキスパート
ゲームフレームワークの設計・開発
現在、Cocos2d-xのゲームの基盤周りを担当しています。
- iOS /Androidアプリを利用したサービスの企画、
設計、開発
- Objective-C、C/C++、Java、Ruby、PHPで開発

アジェンダ

1. iBeaconとは？
2. iBeaconを使ったゲームを作ろう
3. Cocos2d-x x Bluetooth

iBeaconとは？

- iBeaconはApple者の商標
- Bluetooth Low Energyを利用したフレームワーク
- Androidでも4.3からOSで対応
対応機種はまだ少ない
- ビーコンの近接を検出する技術
使い方は規定されていない

iBeaconでできること

iOSではCore Location Frameworkに追加された
また、Multipeer Connectivity Frameworkでも利用
されている

- Beaconの領域観測 (Region)
- Beaconの距離観測 (Ranging)
- iOS7.1からはアプリが起動していなくても機能する。ローカル通知機能に対応

iBeaconを使ったゲームを作ろう

どんなゲームが出来るか？

- iBeacon(Beaconを設置)

宝探し

スタンプラリー

- P2P

すれ違い通信、近くの端末同士のアイテム交換、

バトル、チャットなどが実現可能

必要条件

- Bluetooth Low Energyに対応した機種
iPhone 4S以降、iPad第三世代以降、Nexus 5 /
Nexus 7（新型）
- iOS7以降、Android 4.3以降

ゲームでの利用条件

- 宝探し、スタンプラリー

iOS / Android ○

- P2P

iOS (iOS対抗) ○

Android (Android対抗) ×

Android (セントラル) <-> iOS (ペリフェラル) △

iBeaconを使ったゲームを作りたい（1）

- 宝探しやスタンプラリーは簡単、Android（4.3以降のBLE対応端末）でも動作
しかしBeaconを配置しなくてはならない=>大変
- そこで、今回のゲームはP2Pで作る。
BLEを使うと現状では、AndroidどうしやAndroidとiOSで双方向で接続できない。iOSのみ対応にする。だとCocos2d-xの意味がない。。。。

iBeaconを使ったゲームを作りたい（2）

- なので、今回は、iOS <-> Androidは諦めて、iOS <-> iOSとAndroid <-> AndroidでP2Pに対応したゲームにする
- AndroidはClassic Bluetoothで。。。その代わりほとんどの端末で使える
- 将来、AndroidのBLEでペリフェラルモードが使える様になったらハッピー。

Cocos2d-xでBluetoothの利用（１）

今回は利用価値の高いP2Pに挑戦

- iOS <-> iOS ○
- Android <-> Android ○
- iOS <-> Android ×

Cocos2d-xでBluetoothの利用（2）

- iOSはCore Location、Core Bluetooth、Multipeer Connectivityを使用
- AndroidはClassic Bluetoothを使用
Classic BluetoothとはBLEではないBluetooth 2.1
や3.0など。ペアリングなしで利用
API Level 10（Android 2.3.3）以降

Cocos2d-xでBluetoothの利用（3）

同じアプリのユーザかの認識方法

- iOS: ProximityUUIDをゲーム毎に指定する
- Android: DeviceNameをゲーム毎に指定する

UUIDの生成方法

Mac OS Xの場合

```
$ uuidgen
```

- ターミナルでuuuidgenコマンドで生成

Cocos2d-x Bluetooth P2Pプラグイン

- Cocos2d-x Bluetooth P2Pプラグインを作っています。
- Cocos2d-xからはデバイス(iOS/Android)の意識は当然しない。また、BLEかClassic Bluetoothの意識も必要ない。

Cocos2d-x Bluetooth P2Pプラグインでできること

- 基本は、近くの端末どうしでデータを送受信する。

特にデータのフォーマットは規定していません。文字列なのでJSONとかで通信できます

- Cocos2d-x Bluetooth P2PプラグインはGithubに置きます。(後日)

CCBluetooth

CCBluetooth.h

```
#define RESULT_NOTFOUND_PEER 0
#define RESULT_RECEIVE_MESSAGE 1

#define STATUS_ERROR -1
#define STATUS_OK 1

class CCBluetoothDelegate
{
public:
    virtual void onResult(int resultCode, int status, const char *error, const char *peerID, const char *message) {};
};

class CCBluetooth
{
public:
    CCBluetooth(CCBluetoothDelegate* delegate);
    ~CCBluetooth();

    void start(const char *peerID, const char *message);
    void stop();

private:
    CCBluetoothDelegate *_delegate;
};
```

CCBluetooth

Function

// 開始

```
void start(  
    const char *peerID, // ユーザ(端末)を識別  
    const char *message // 接続後に送るデータ  
);
```

// 終了

```
void stop();
```

CCBluetooth

CCBluetoothDelegate

```
void onResult(  
    int resultCode, // 結果  
    int status, // ステータス  
    const char *error, // エラー内容  
    const char *peerID, // 相手のID  
    const char *message // 受け取ったデータ  
);
```

サンプルコード

SampleScene.cpp

```
void SampleScene::onEnter() {
    cocos2d::CCLayer::onEnter();
    bluetooth = new bluetooth_plugin::CCBluetooth(this);
    bluetooth->start("<PeerID>", "<message>");
}

void SampleScene::onExit() {
    cocos2d::CCLayer::onExit();
    bluetooth->stop("<PeerID>", "<message>");
}

// CCBluetoothDelegate
void SampleScene::onResult(
    int resultCode,
    int status,
    const char *error,
    const char *peerID,
    const char *message) {
    CCLOG("相手: %s", peerID.c_str());
    CCLOG("メッセージ: %s", message.c_str());
}
```

注意点

- 通信データが暗号化されないため、必要に応じてアプリで暗号化を行ってください。

資料やコードはGitHub置きます。

GitHub: <https://github.com/tomookaku>

Gist: <https://gist.github.com/tomookaku>

facebook: <https://www.facebook.com/tomoo.kaku>