Dent bleue pour iOS & OS X

@stuffmc  Apple Technologist

[Square Bracket Gang]
Dent bleue pour iOS & OS X

@stuffmc Apple Technologist

[Return -> Gang]
Since iOS 5 and OS X 10.7 — Universal API
Since iPhone 4S, iPad 3, iPod touch 5th gen
Bluetooth hardware makers have their own CB* Wrapper
Watch probably has BTLE*, but who cares, really?
Apple ❤ that ❤ Rate Sensor for its demos
```swift
central = CBCentralManager(delegate: self, queue: nil)

func centralManagerDidUpdateState(central: CBCentralManager!) {
    if central.state == .PoweredOn {
        central?.scanForPeripheralsWithServices(nil, options: nil)
    }
}

func centralManager(central: CBCentralManager!,
didDiscoverPeripheral peripheral: CBPeripheral!,
RSSI: NSNumber!)
{
    central.connectPeripheral(peripheral, options: nil)
    central.stopScan() // DO NOT FORGET to not drain the user’s battery!
}
```

**WARNING!** nil → All Peripherals
Client

Wants data

Server

Has data

Central

Peripheral

Core Bluetooth
Core Bluetooth

@stuffmc

iOS 6 & OS X 10.9
Mac and iOS devices can function as peripherals

Service
Heart Rate Monitor

Characteristic
Heart Rate Measurement

Characteristic
Body Sensor Location

Server
Has data

Peripheral
func centralManager(central: CBCentralManager!,
               didConnectPeripheral peripheral: CBPeripheral!)
{
    peripheral.delegate = self
    peripheral.discoverServices(nil)
}

func peripheral(peripheral: CBPeripheral!,
               didDiscoverServices error: NSError!)
{
    for service in peripheral.services {
        peripheral.discoverCharacteristics(nil,
                                             forService: service as CBService)
    }
}

WARNING! nil → All Services, All Characteristics
func peripheral(didDiscoverCharacteristicsForService service: CBService!)
{
    for char in service.characteristics {
        peripheral.writeValue(data, forCharacteristic: char, type: .WithoutResponse)
            // e.g. “Turn LED ON” (sending 0x1)
        peripheral.readValueForCharacteristic(characteristic: char)
            // “One Time” Read. e.g. Body Sensor Location
        peripheral.setNotifyValue(true, forCharacteristic: char)
            // subscribe to “didUpdateValueForCharacteristic” e.g. Heart Rate Measurement
    }
}

func peripheral(peripheral: CBPeripheral!,
    didUpdateValueForCharacteristic characteristic: CBCharacteristic!)
{
    ... 
}
Core Bluetooth

Client

Wants data

Server

Has data

Central

Peripheral
UART: Old Skool Byte Communication
You might have a single service & characteristic, that simple!

```swift
uartTxChar: CBCharacteristic?
uartRxChar: CBCharacteristic?

peripheral.writeUARTData(data)

// Under the hood
peripheral.writeValue(data, forCharacteristic: uartRxChar, type: .WithoutResponse)

// Easy “receive”
func didReceiveUARTData(data: NSData!) {
```
✓ Your Bible: Core Bluetooth Overview

✓ Remember: Bluetooth chip or device makers might have their own CB Wrapper for you! Check this first!!!

✓ You might not need to worry about “Filtering” the noise

✓ Sadly, many things are still Classic Bluetooth

✓ CB only does BTLE

✓ The future of 🍌 + Bluetooth?