

Advanced Core Bluetooth

Session 705

Joakim Linde

Senior Wireless Architect

These are confidential sessions—please refrain from streaming, blogging, or taking pictures

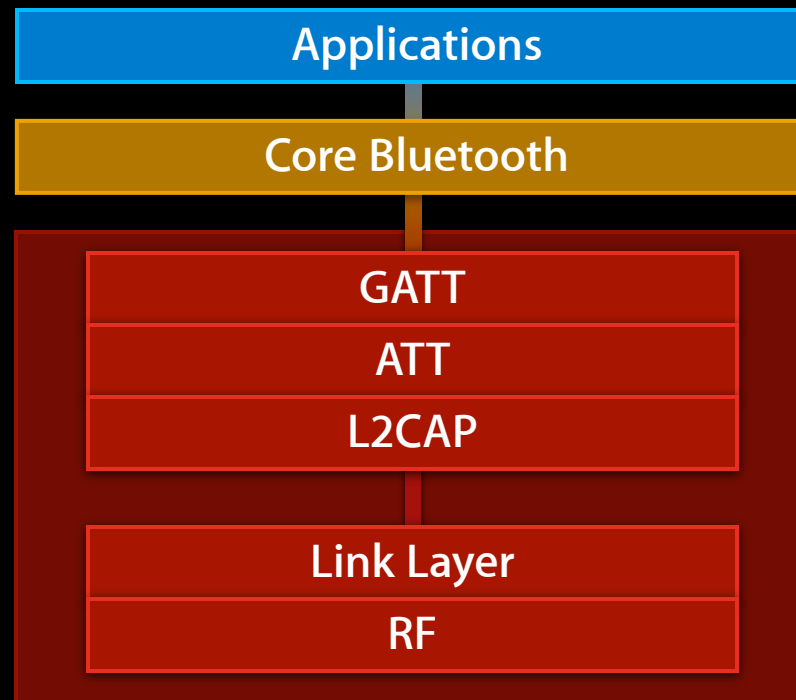
Introduction

- Core Bluetooth
 - Introduced in iOS 5 and OS X (10.7)
 - New functionality in iOS 6

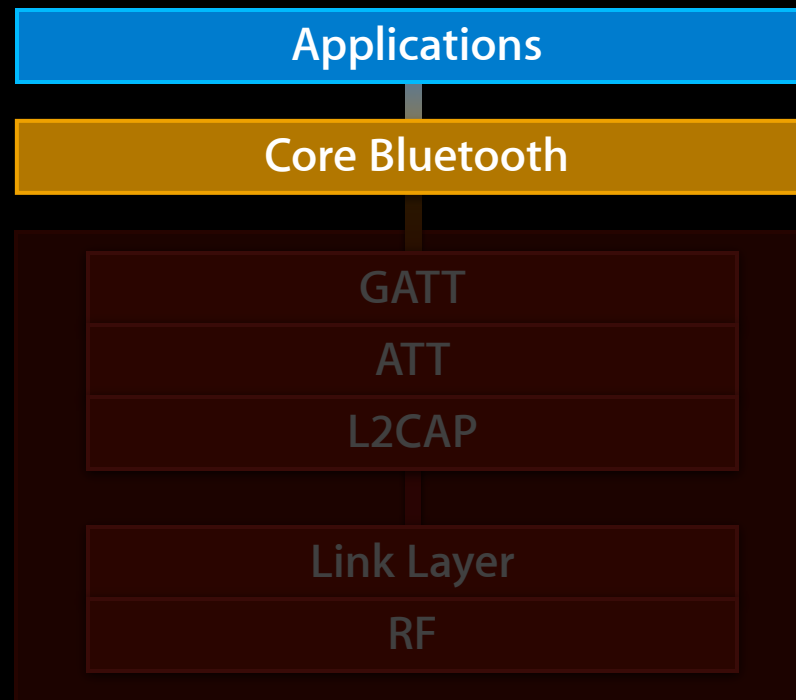
What You Will Learn

- Discover Core Bluetooth
 - Object model
 - Central and client
 - Peripheral and server
- Tips and reminders
 - iOS backgrounding
 - Discovering devices efficiently
 - Pairing
- Demo

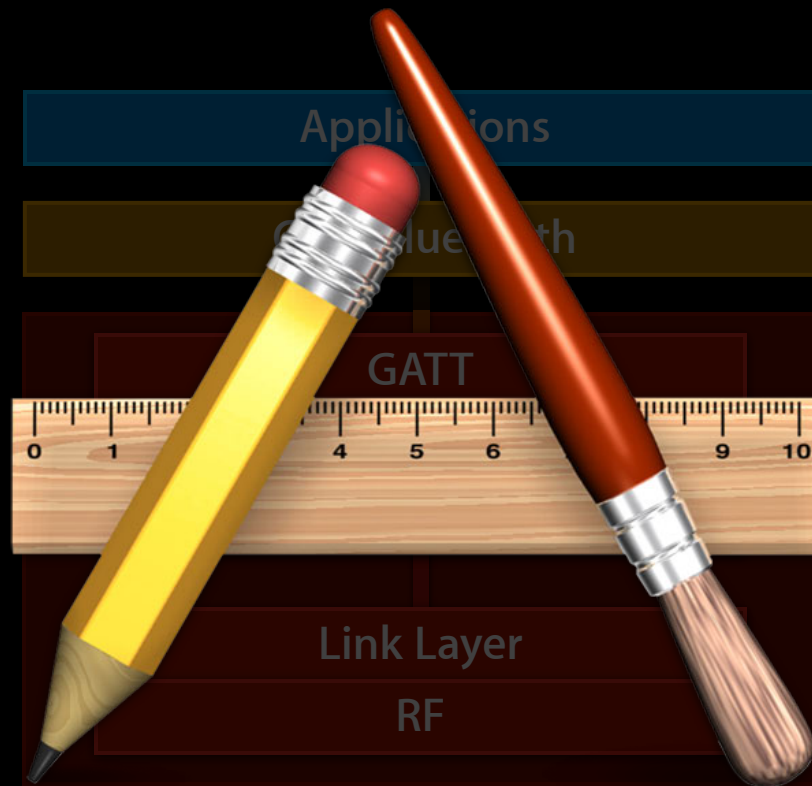
Technology Framework



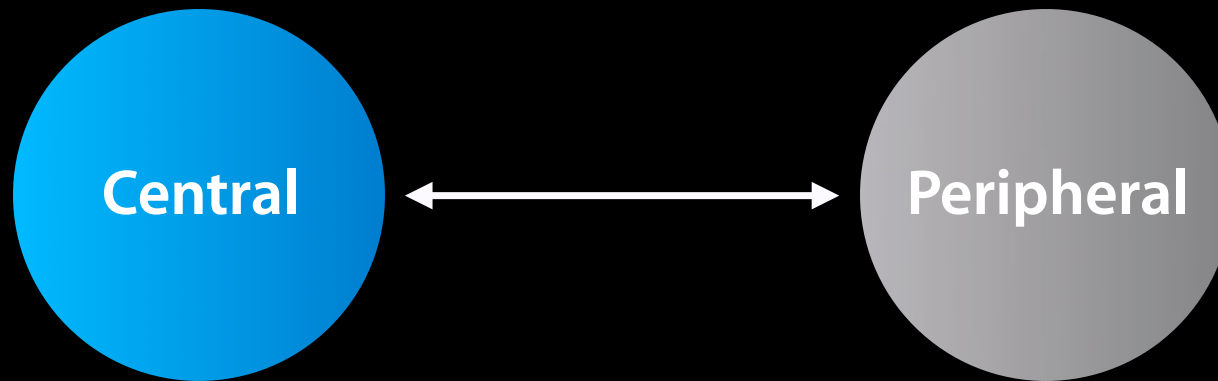
Technology Framework



Technology Framework

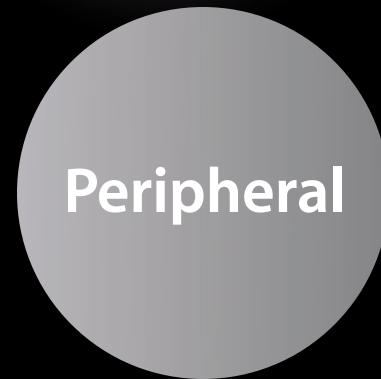


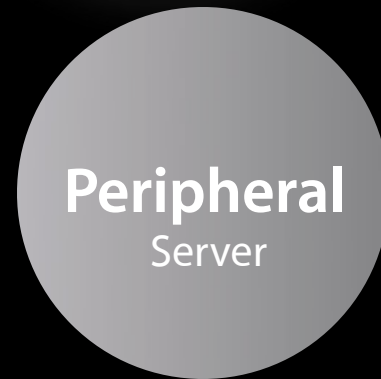
It's all about the app



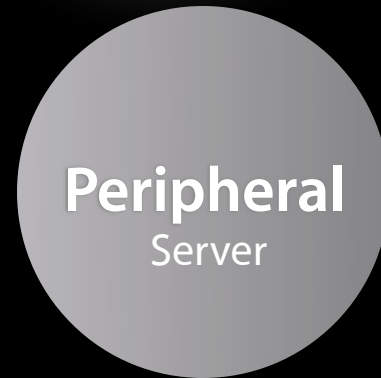


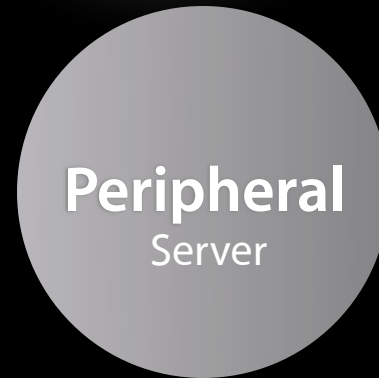




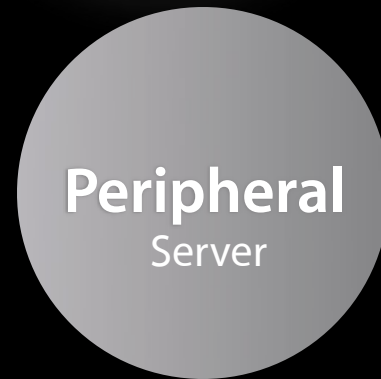








CBCentralManager

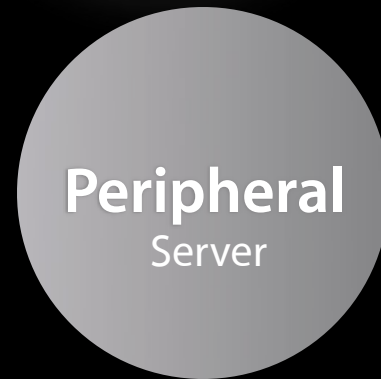


CBCentralManager

CBPeripheral



Central
Client

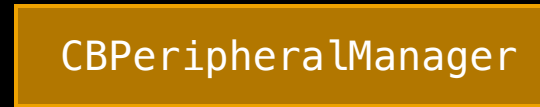


Peripheral
Server

CBCentralManager

CBPeripheral

CBPeripheralManager

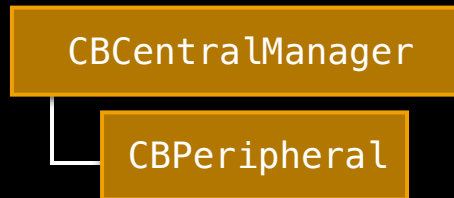


Core Bluetooth Object Model

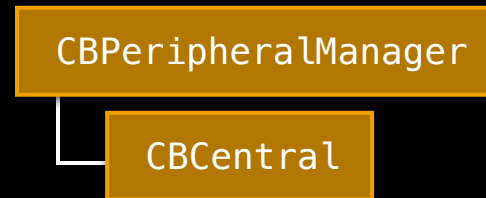


Main Objects

New in iOS 5



New in iOS 6



Core Bluetooth Object Model



New in iOS 5

Main Objects

CBCentralManager

CBPeripheral

Data Objects

CBService

CBCharacteristic

New in iOS 6

CBPeripheralManager

CBCentral

CBMutableService

CBMutableCharacteristic

Core Bluetooth Object Model



New in iOS 5

New in iOS 6

Main Objects

CBCentralManager

CBPeripheralManager

CBPeripheral

CBCentral

Data Objects

CBService

CBMutableService

CBCharacteristic

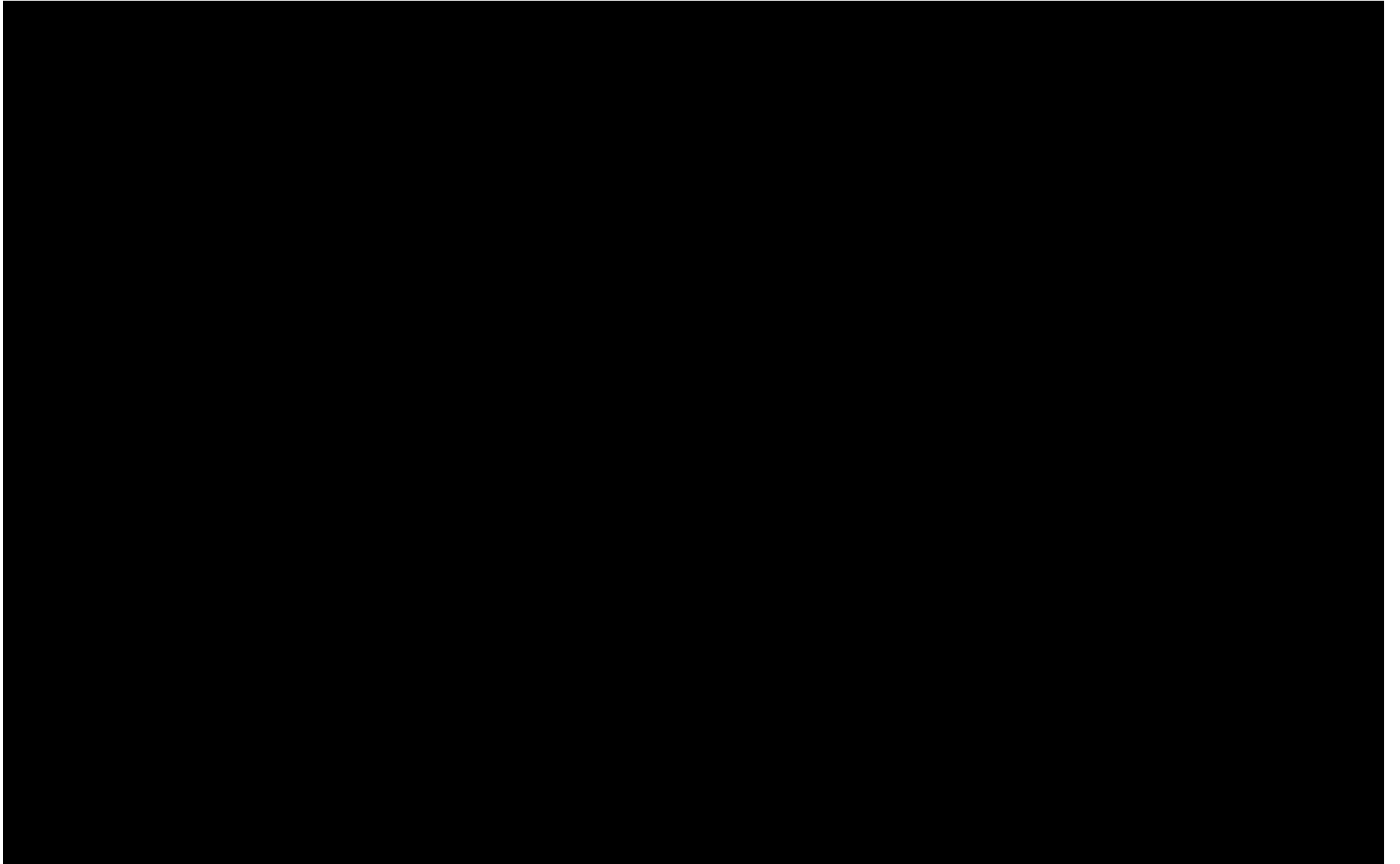
CBMutableCharacteristic

Helper Objects

CBUUID

CBATTRequest

Central and Client



1

Discover

2

Explore

3

Interact

Client Side—3 Easy Steps

Discover

Discover



Discover



Discover



Discover

Peripheral



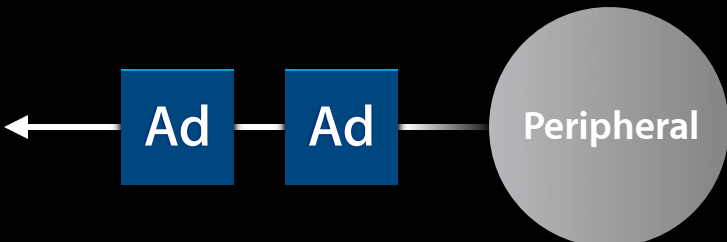
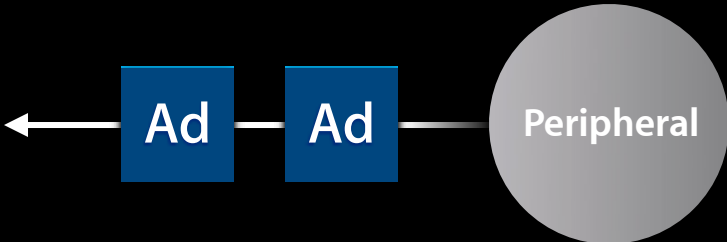
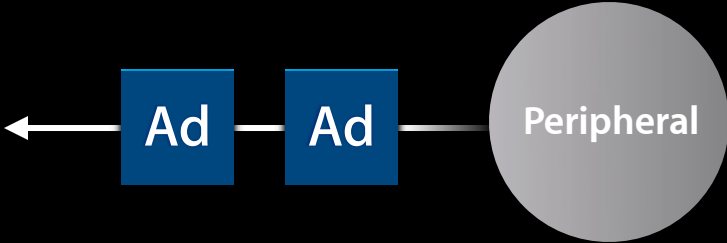
Peripheral



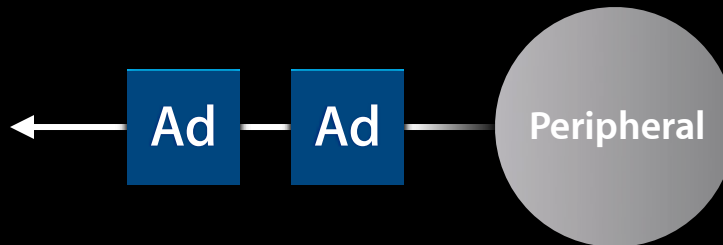
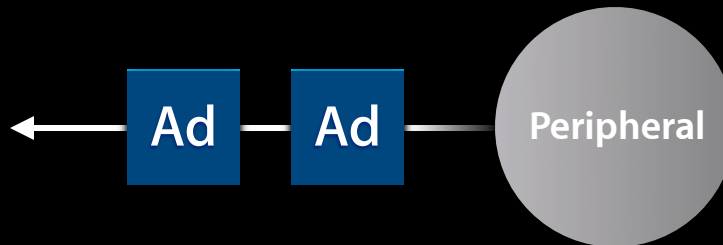
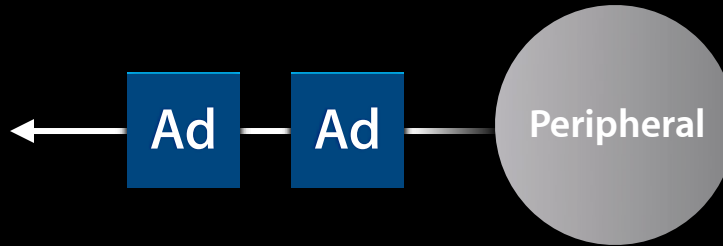
Peripheral



Discover



Discover



Discover and Connect

CBCentralManager

<CBCentralManagerDelegate>

Discover and Connect

CBCentralManager

<CBCentralManagerDelegate>



Discover and Connect

CBCentralManager

<CBCentralManagerDelegate>

scanForPeripheralsWithServices



Discover and Connect

CBCentralManager

<CBCentralManagerDelegate>

scanForPeripheralsWithServices

didDiscoverPeripheral
didDiscoverPeripheral
didDiscoverPeripheral



Discover and Connect

CBCentralManager

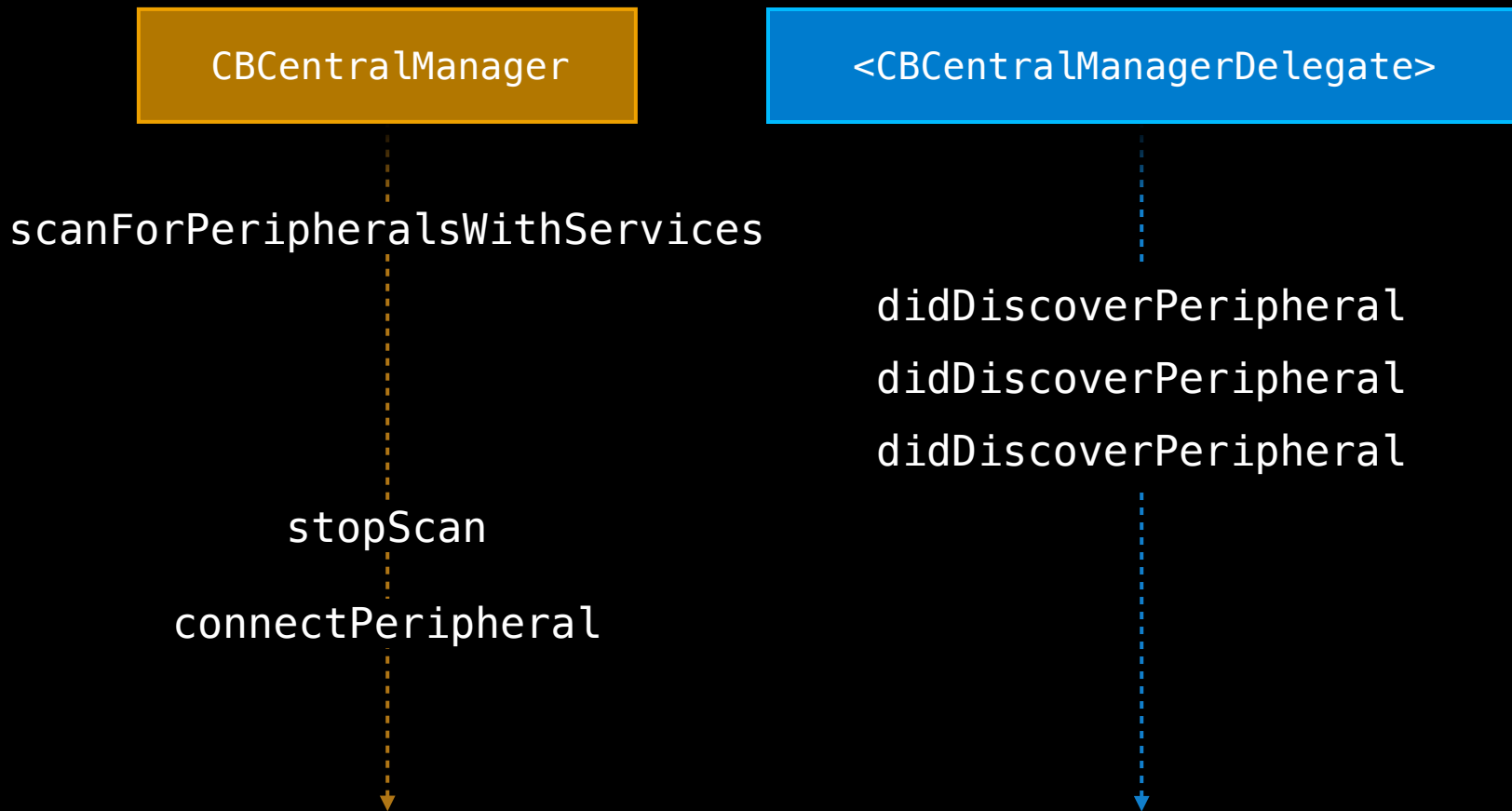
<CBCentralManagerDelegate>

scanForPeripheralsWithServices

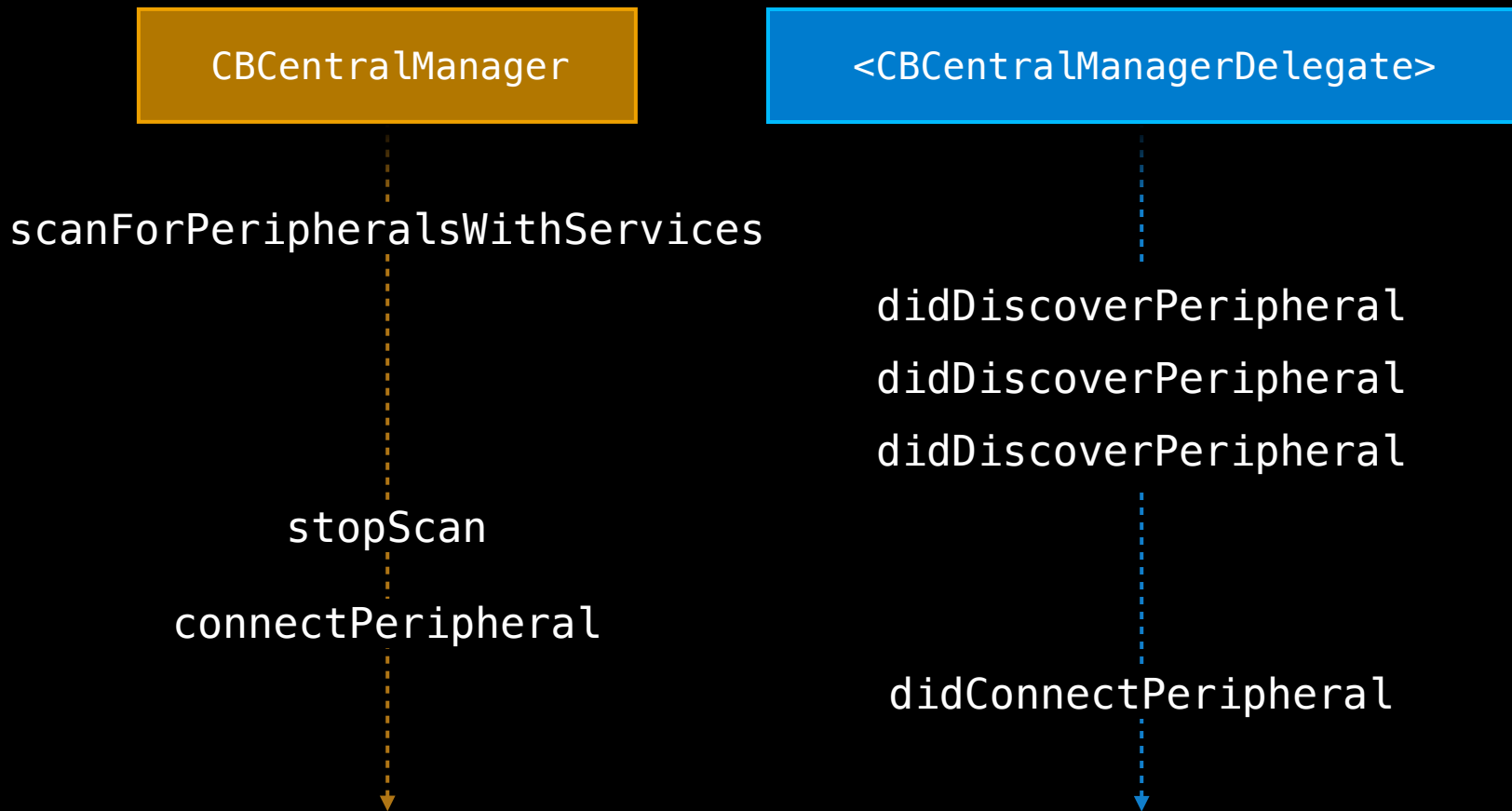
didDiscoverPeripheral
didDiscoverPeripheral
didDiscoverPeripheral

stopScan

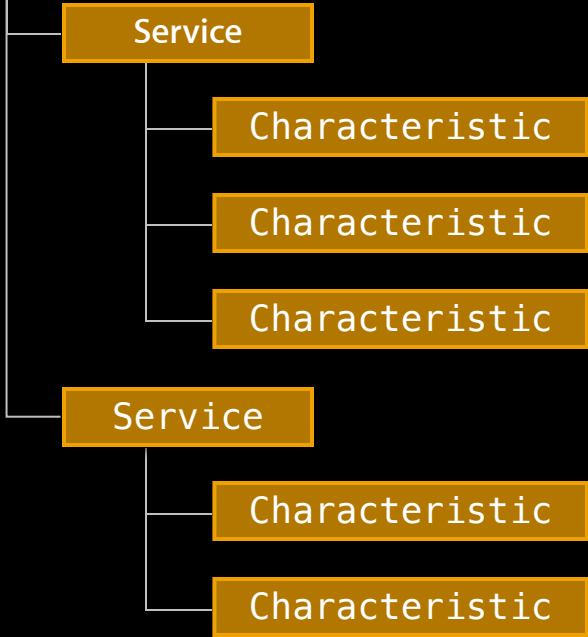
Discover and Connect



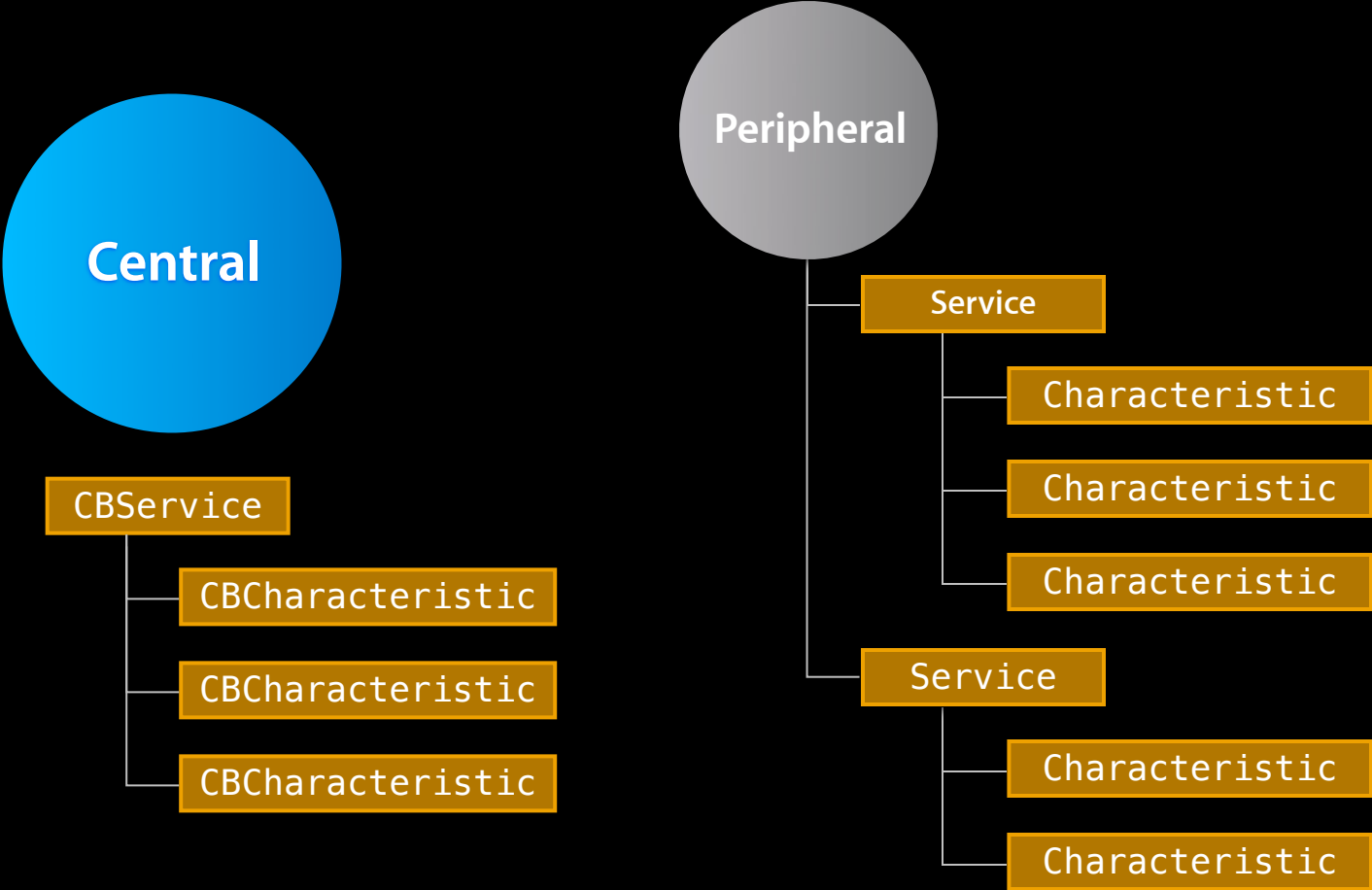
Discover and Connect



Explore



Explore



Example



Server

Service
Heart Rate Monitor

Characteristic
Heart Rate Measurement

Characteristic
Body Sensor Location

Explore

CBPeripheral

<CBPeripheralDelegate>

Explore

CBPeripheral

<CBPeripheralDelegate>



Explore

CBPeripheral

<CBPeripheralDelegate>

discoverServices



Explore

CBPeripheral

discoverServices



<CBPeripheralDelegate>

didDiscoverServices



Explore

CBPeripheral

<CBPeripheralDelegate>

discoverServices

didDiscoverServices

discoverCharacteristics:forService



Explore

CBPeripheral

<CBPeripheralDelegate>

discoverServices

didDiscoverServices

discoverCharacteristics:forService

didDiscoverCharacteristicsForService



Interact

CBPeripheral

<CBPeripheralDelegate>

Interact

CBPeripheral

<CBPeripheralDelegate>



Interact

CBPeripheral

<CBPeripheralDelegate>

readValueForCharacteristic



Interact

CBPeripheral

readValueForCharacteristic



<CBPeripheralDelegate>

didUpdateValueForCharacteristic



Interact

CBPeripheral

<CBPeripheralDelegate>

readValueForCharacteristic

didUpdateValueForCharacteristic



Interact

CBPeripheral

<CBPeripheralDelegate>

readValueForCharacteristic

didUpdateValueForCharacteristic

setNotifyValue:forCharacteristic



Interact

CBPeripheral

<CBPeripheralDelegate>

readValueForCharacteristic

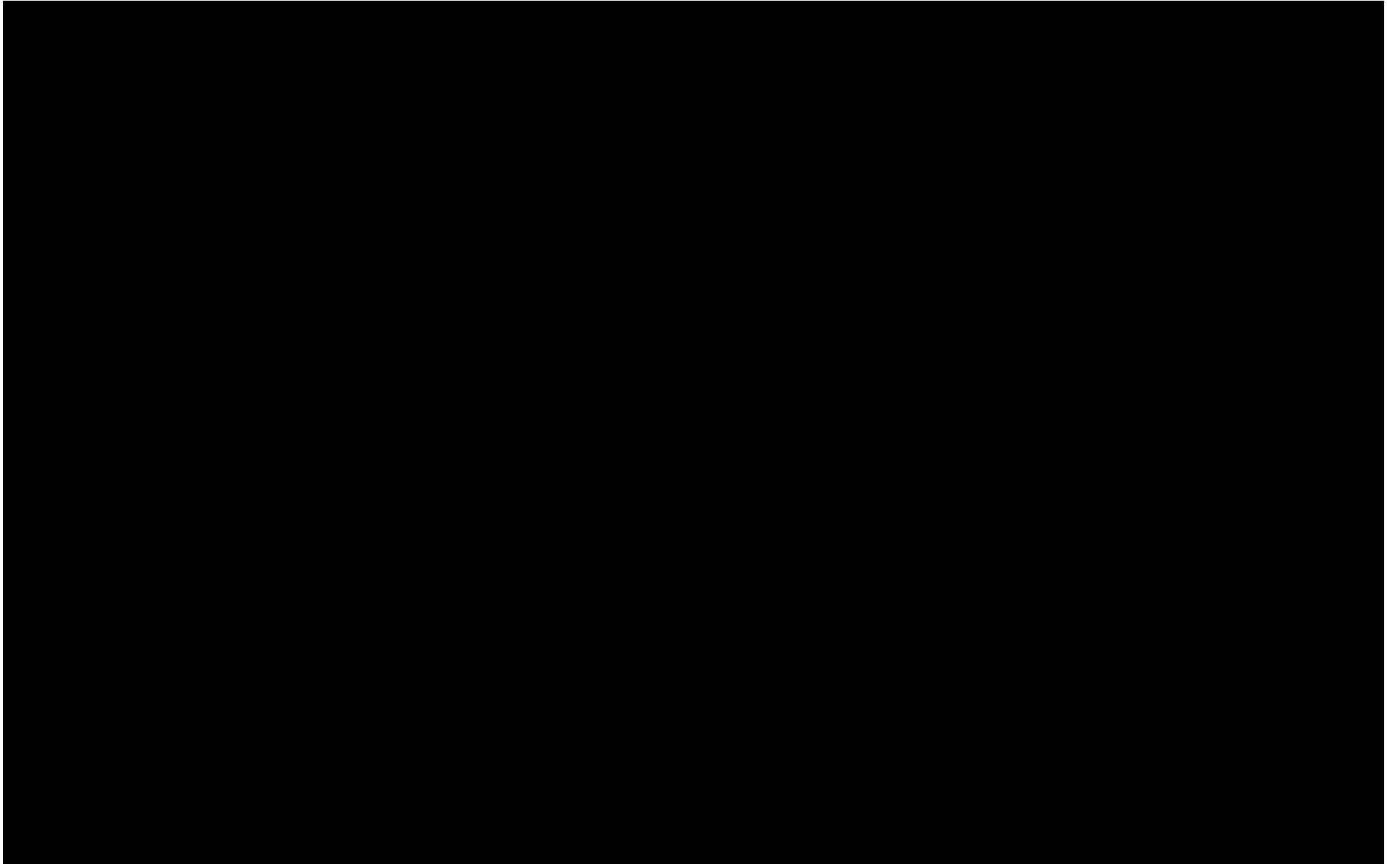
didUpdateValueForCharacteristic

setNotifyValue:forCharacteristic

didUpdateValueForCharacteristic
didUpdateValueForCharacteristic
didUpdateValueForCharacteristic



Reconnecting



1
Scan and
Connect

2
Reconnect
Using UUID

3
Already
Connected?

Reconnect—3 Ways

Reconnect Using UUID

CBCentralManager

<CBCentralManagerDelegate>

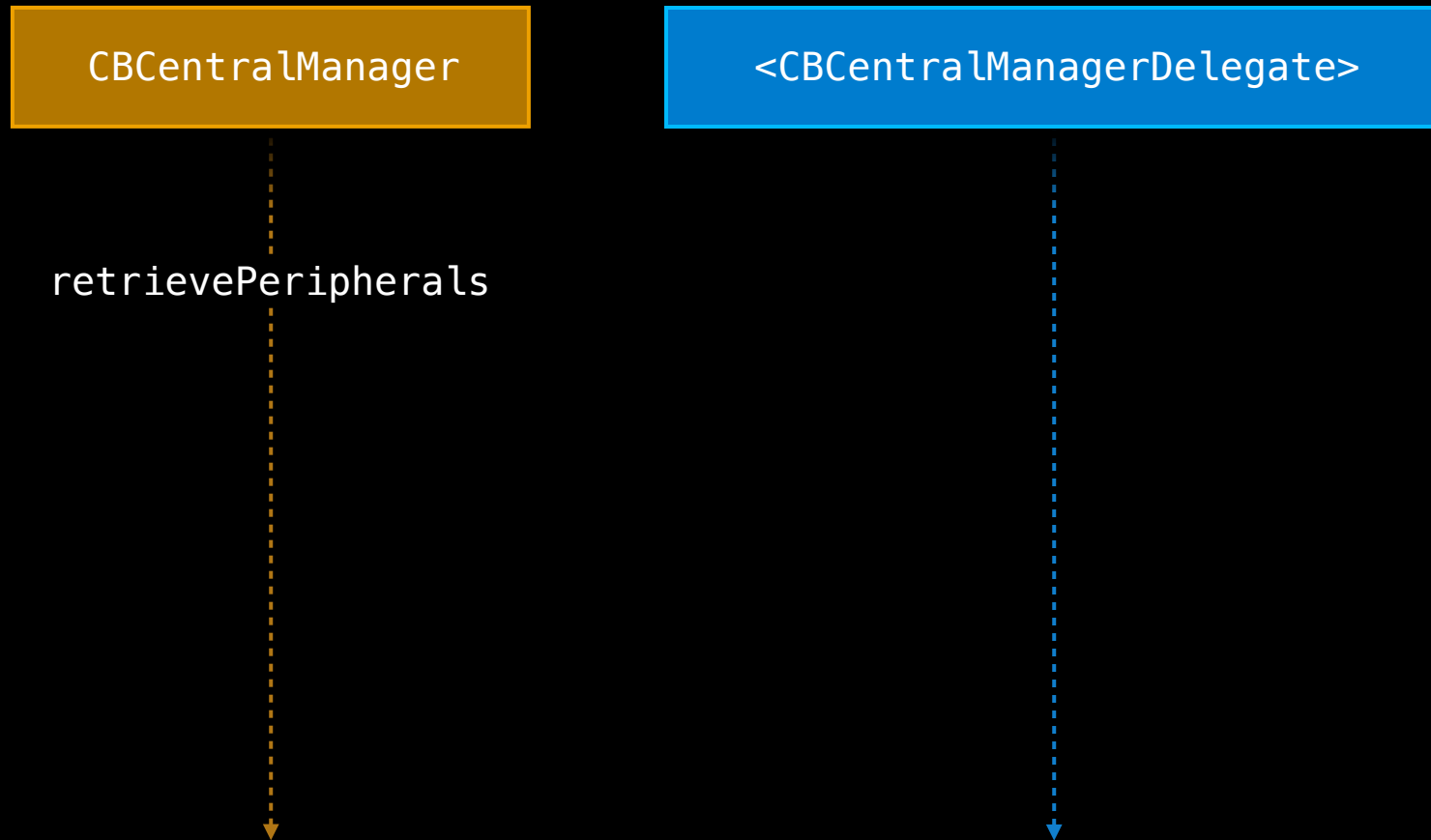
Reconnect Using UUID

CBCentralManager

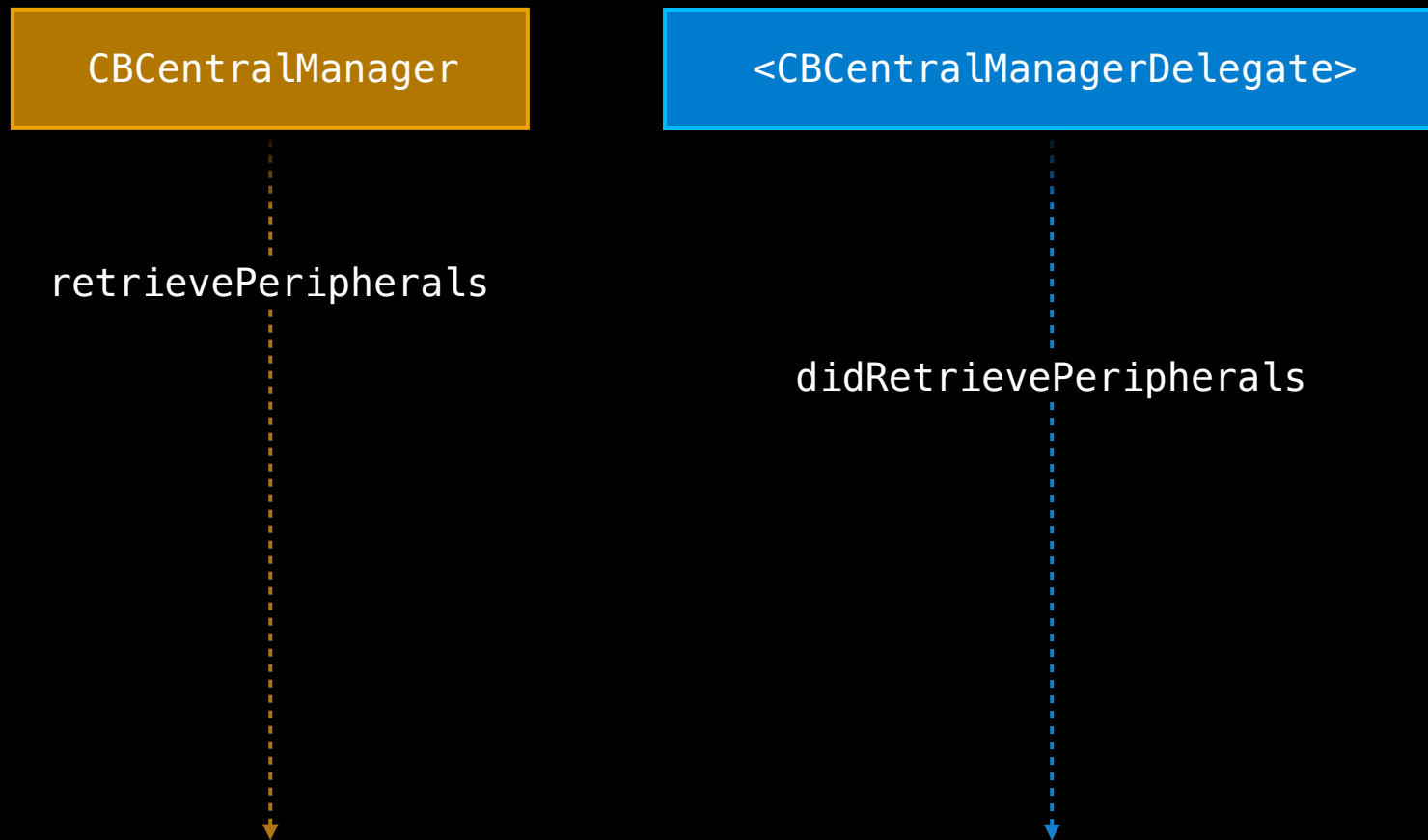
<CBCentralManagerDelegate>



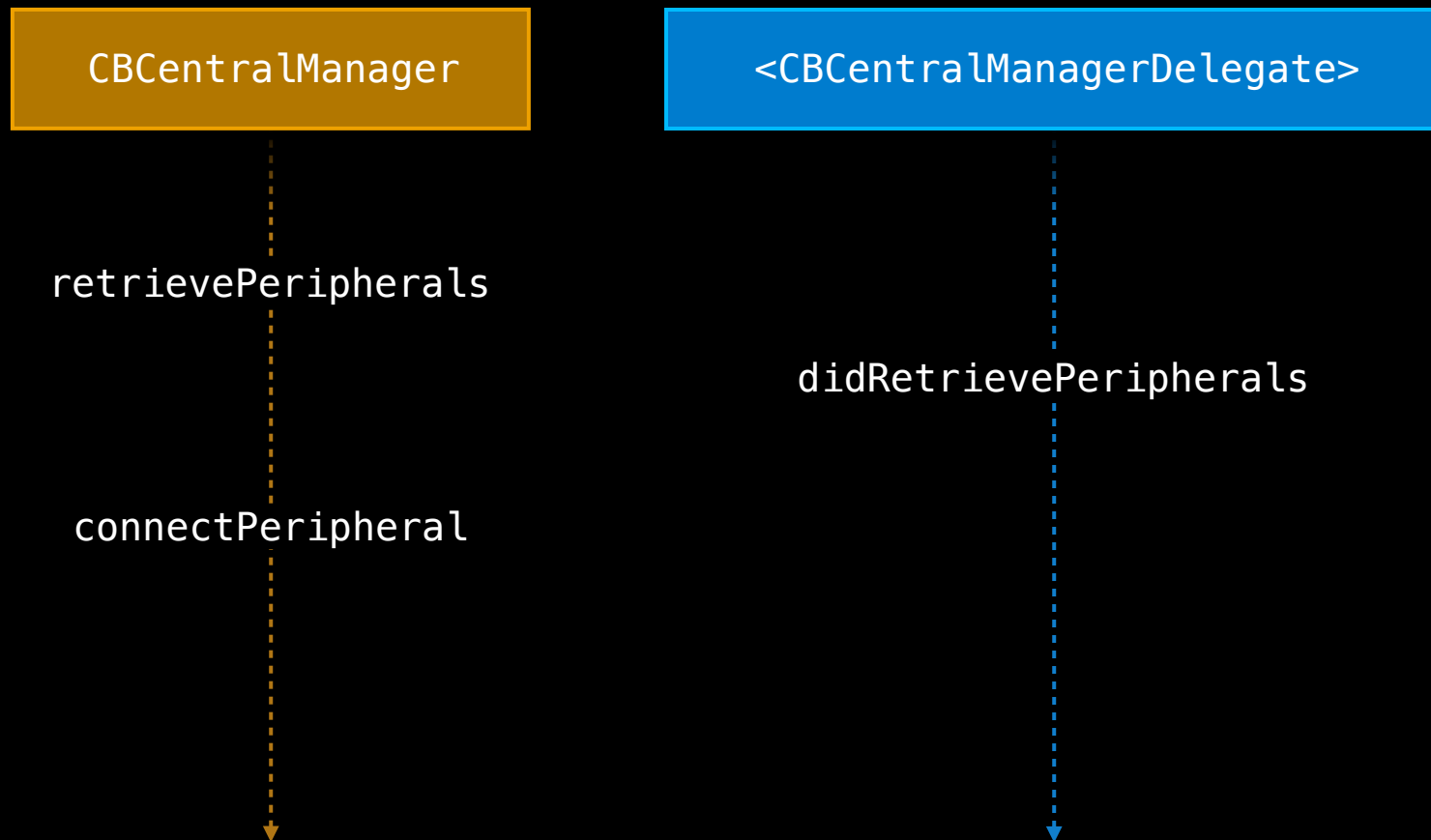
Reconnect Using UUID



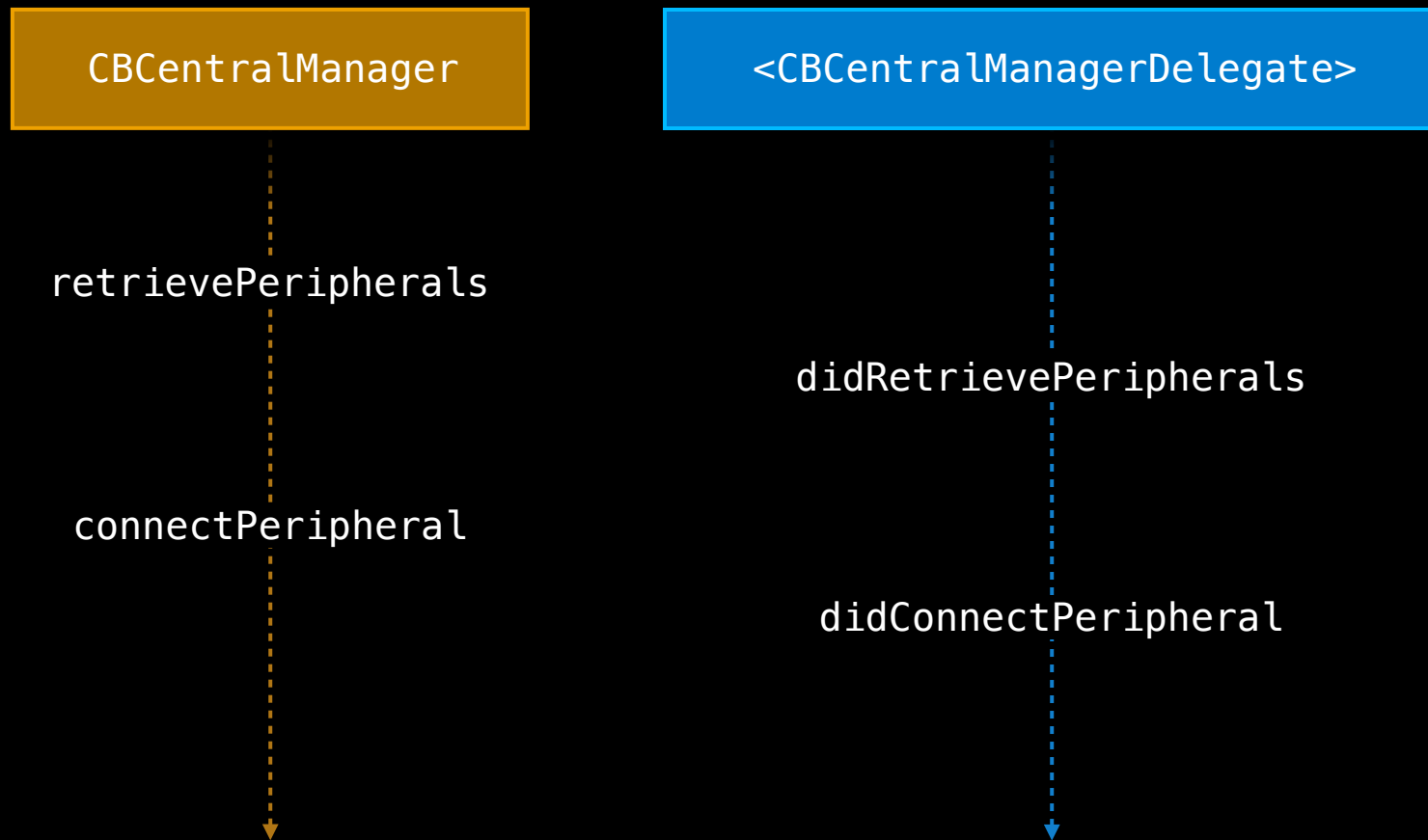
Reconnect Using UUID

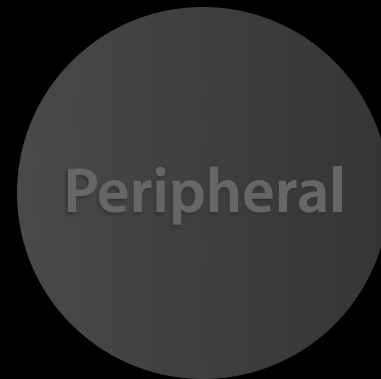


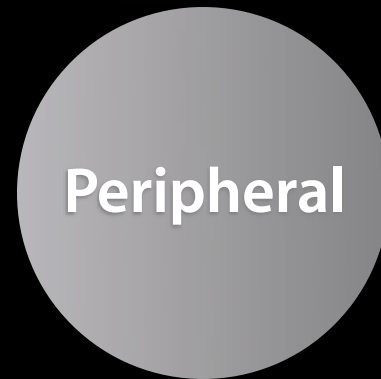
Reconnect Using UUID



Reconnect Using UUID





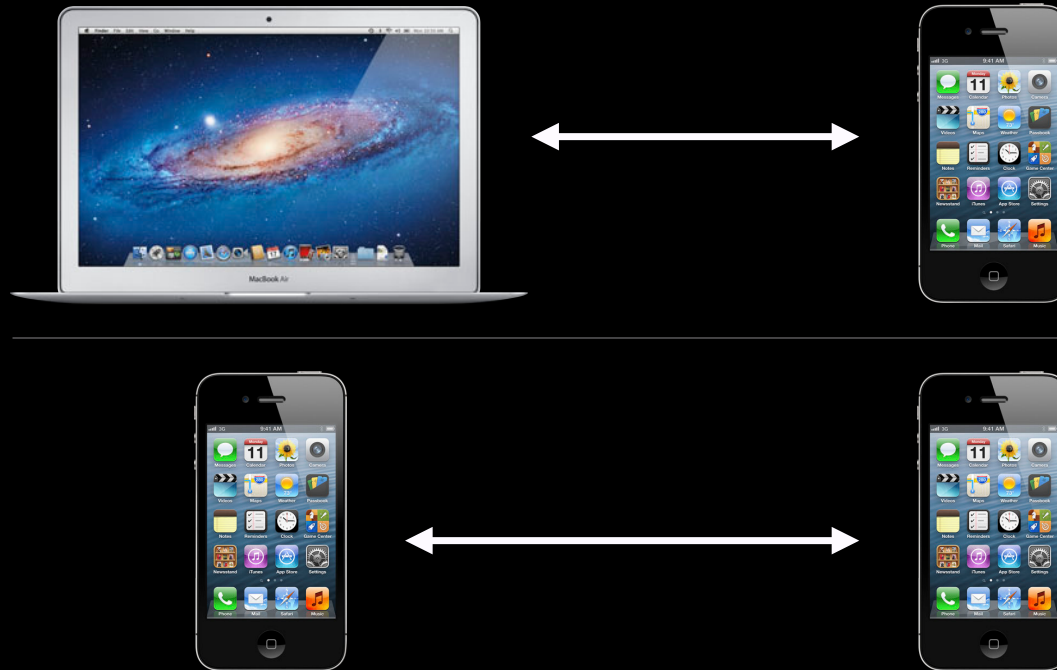


Peripheral and Server



New





While app is backgrounded

1
Build
Tree

2
Publish
and Advertise

3
Handle
Requests

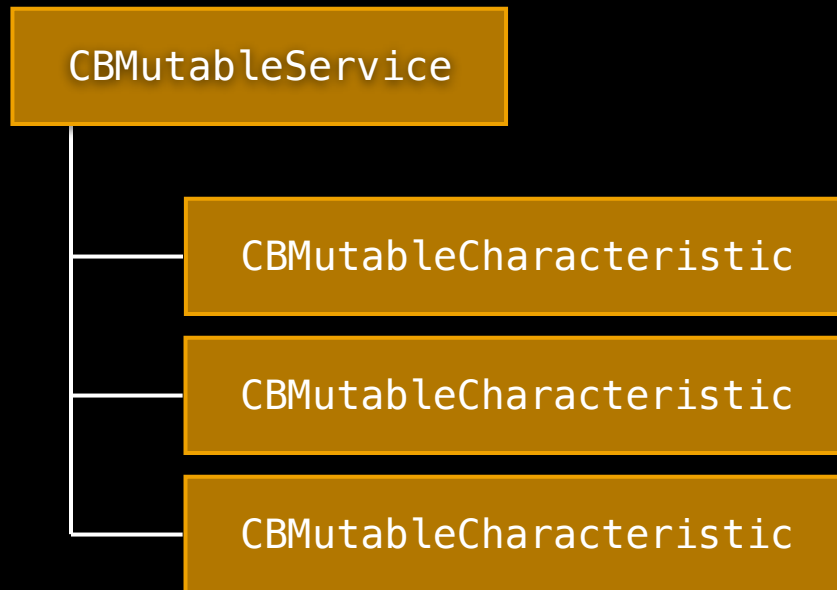
1
**Build
Tree**

2
**Publish
and Advertise**

3
**Handle
Requests**

Server Setup—3 Easy Steps

Build Tree



Variables and Data

Variables and Data

Static Data

NSMutableDictionary

NSData



Variables and Data

Static Data



Dynamic Data





Publishing and Advertise

CBPeripheralManager

<CBPeripheralManagerDelegate>

Publishing and Advertise

CBPeripheralManager

<CBPeripheralManagerDelegate>



Publishing and Advertise

CBPeripheralManager

<CBPeripheralManagerDelegate>

addService



Publishing and Advertise

CBPeripheralManager

addService



<CBPeripheralManagerDelegate>

didAddService



Publishing and Advertise

CBPeripheralManager

<CBPeripheralManagerDelegate>

addService

didAddService

startAdvertising



Publishing and Advertise

CBPeripheralManager

<CBPeripheralManagerDelegate>

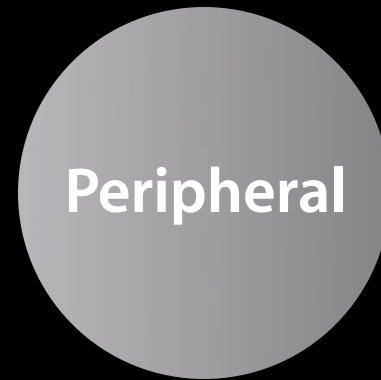
addService

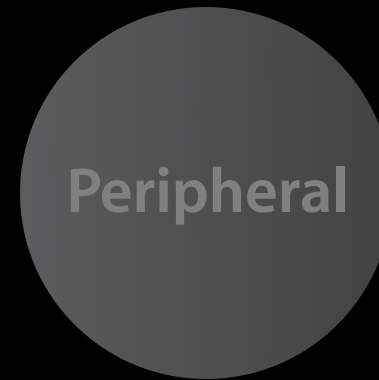
didAddService

startAdvertising

didStartAdvertising















Explore



Dynamic Data

CBPeripheralManager

<CBPeripheralManagerDelegate>

Dynamic Data

CBPeripheralManager

<CBPeripheralManagerDelegate>



Dynamic Data

CBPeripheralManager

<CBPeripheralManagerDelegate>

centralDidConnect



Dynamic Data

CBPeripheralManager



<CBPeripheralManagerDelegate>

centralDidConnect

didReceiveReadRequest



Dynamic Data

CBPeripheralManager

respondToRequest



<CBPeripheralManagerDelegate>

centralDidConnect

didReceiveReadRequest



Dynamic Data

CBPeripheralManager

respondToRequest



<CBPeripheralManagerDelegate>

centralDidConnect

didReceiveReadRequest

centralDidDisconnect



Subscribing

CBPeripheralManager

<CBPeripheralManagerDelegate>

Subscribing

CBPeripheralManager

<CBPeripheralManagerDelegate>



Subscribing

CBPeripheralManager

<CBPeripheralManagerDelegate>

centralDidConnect



Subscribing

CBPeripheralManager

<CBPeripheralManagerDelegate>

centralDidConnect

didSubscribeToCharacteristic



Subscribing

CBPeripheralManager

<CBPeripheralManagerDelegate>

updateValue
updateValue
updateValue

centralDidConnect
didSubscribeToCharacteristic



Subscribing

CBPeripheralManager

updateValue
updateValue
updateValue

<CBPeripheralManagerDelegate>

centralDidConnect
didSubscribeToCharacteristic

centralDidDisconnect





App backgrounded

Tips and Reminders

iOS Backgrounding



- Backgrounded apps
 - Scan and advertise
 - Connect
 - Interact



iOS Backgrounding



- Backgrounded apps
 - Scan and advertise
 - Connect
 - Interact
- “Start and stop” concept



iOS Backgrounding



- Backgrounded apps
 - Scan and advertise
 - Connect
 - Interact
- “Start and stop” concept
- Info.plist:

```
UIBackgroundModes =  
    [ "bluetooth-central" ]
```



Be Mindful



Discovering Devices



- Foreground app
 - High-intensity scanning
- Background app/screen off
 - Low-intensity scanning



Discovering Devices



- Foreground app
 - High-intensity scanning
- Background app/screen off
 - Low-intensity scanning



Background mode → longer discovery times

Short Advertising Intervals



Short Advertising Intervals

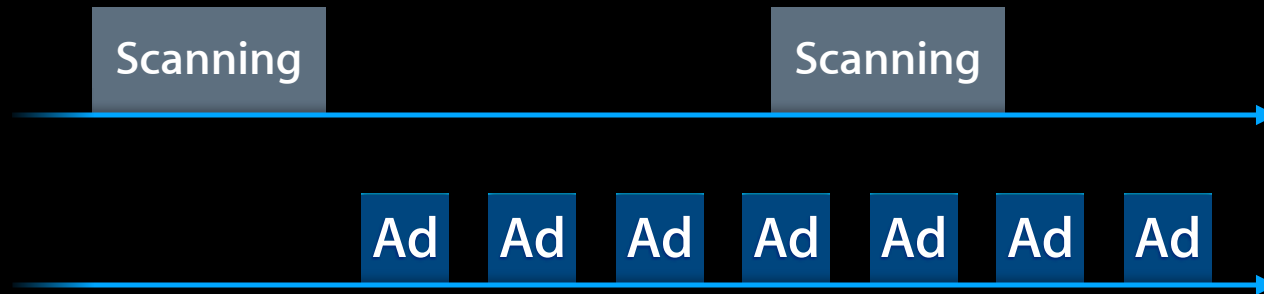


Short advertising interval → Being seen

Short Advertising Intervals



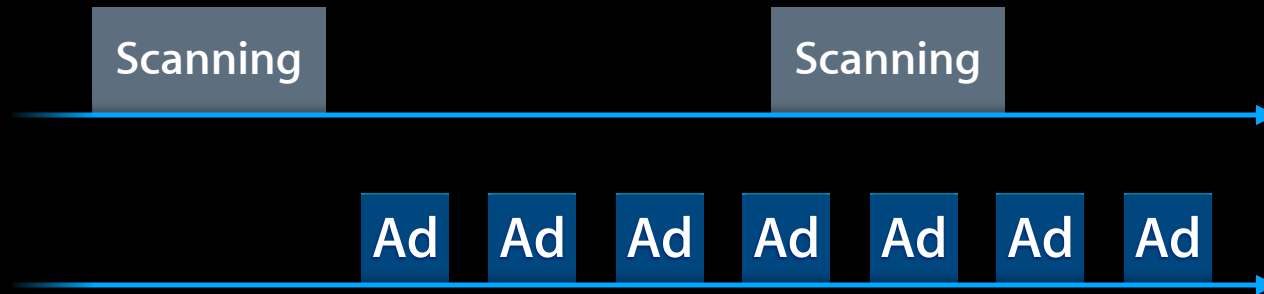
Short advertising interval → Being seen



Short Advertising Intervals



Short advertising interval → Being seen



Do high intensity advertising

Advertising Packet



Advertising Packet



Ad

Advertising Packet



Ad

Name

Advertising Packet



Ad

Name
Tx Power

Advertising Packet



Ad

Name
Tx Power
Service UUID

Advertising Packet



Ad

Name
Tx Power
Service UUID
Service Data

Advertising Packet



Ad

Name
Tx Power
Service UUID
Service Data

Max: 31 bytes

Got UUID?



- 16-bit UUID
 - Assigned by the Bluetooth SIG
 - <http://developer.bluetooth.org>
- 128-bit UUID
 - Assigned by **you!**

Got UUID?



- 16-bit UUID
 - Assigned by the Bluetooth SIG
 - <http://developer.bluetooth.org>
- 128-bit UUID
 - Assigned by **you!**

```
joakim — bash — 164x45
bash
$ uuidgen
43ECBC16-07ED-43ECBC16272-9398-784868AB1751
$
```


Connection Interval



- Initial connection interval
 - Short!
 - Peripheral: Change?
- Determines latency
- See Bluetooth guidelines for rules regarding Connection Parameters



Connection Interval



- Initial connection interval
 - Short!
 - Peripheral: Change?
- Determines latency
- See Bluetooth guidelines for rules regarding Connection Parameters

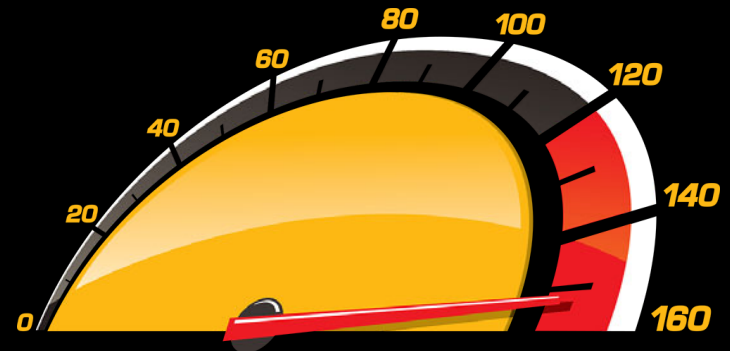


<http://developer.apple.com/hardware/drivers/BluetoothDesignGuidelines.pdf>

Speeding up Connections



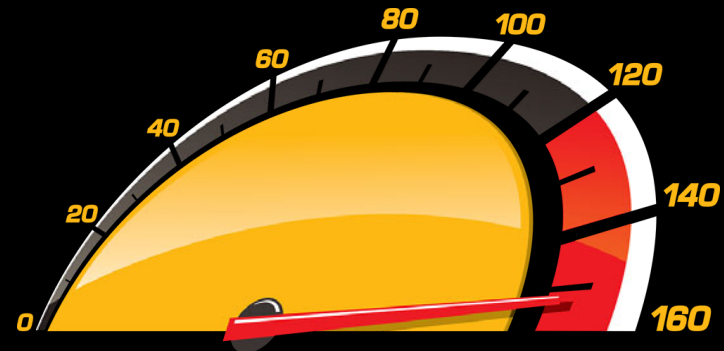
- Central caches
 - Discovered Services
 - Discovered Characteristics
- Values are **not** cached
- Only for paired Peripherals!



Speeding up Connections



- Central caches
 - Discovered Services
 - Discovered Characteristics
- Values are **not** cached
- Only for paired Peripherals!



Fully transparent to the app

Pairing

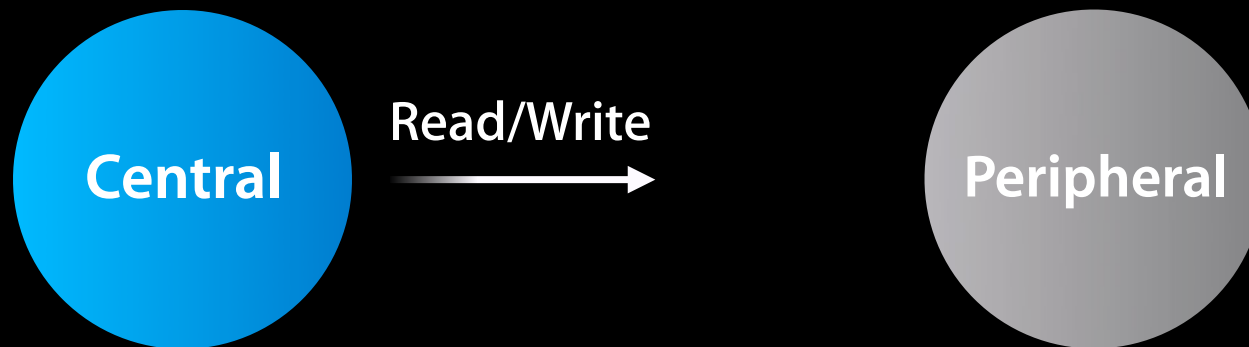
- Peripheral triggers Pairing
 - By rejecting a read/write
- Paired → Encrypted link



Pairing



- Peripheral triggers Pairing
 - By rejecting a read/write
- Paired → Encrypted link



Pairing



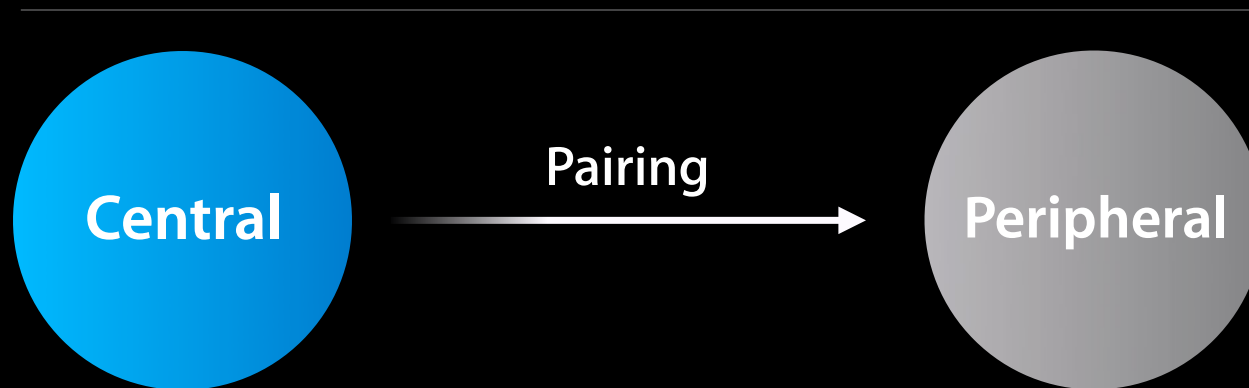
- Peripheral triggers Pairing
 - By rejecting a read/write
- Paired → Encrypted link



Pairing



- Peripheral triggers Pairing
 - By rejecting a read/write
- Paired → Encrypted link



Pairing



- Peripheral triggers Pairing
 - By rejecting a read/write
- Paired → Encrypted link



Pairing



- Peripheral triggers Pairing
 - By rejecting a read/write
- Paired → Encrypted link



Is BT LE Available?



CBCentralManager

delegate	CBCentralManagerDelegate
state	CBCentralManagerState

Is BT LE Available?



CBCentralManager

delegate CBCentralManagerDelegate
state CBCentralManagerState

← **Check state before using**
Make sure BT LE is supported and
turned on before using the Core
Bluetooth framework

Manager States



Central/Peripheral manager can be in one of the following states:

State	Value	Description
Unknown	<code>CBCentralManagerStateUnknown</code>	State unknown; update imminent
Resetting	<code>CBCentralManagerStateResetting</code>	Connection with system momentarily lost
Unsupported	<code>CBCentralManagerStateUnsupported</code>	Platform does not support BT LE
Unauthorized	<code>CBCentralManagerStateUnauthorized</code>	App not authorized to use BT LE
Powered off	<code>CBCentralManagerStatePoweredOff</code>	Bluetooth is powered off
Powered on	<code>CBCentralManagerStatePoweredOn</code>	Bluetooth is powered on and available to use

Manager States



Central/Peripheral manager can be in one of the following states:

State	Value	Description
Unknown	<code>CBCentralManagerStateUnknown</code>	State unknown; update imminent
Resetting	<code>CBCentralManagerStateResetting</code>	Connection with system momentarily lost
Unsupported	<code>CBCentralManagerStateUnsupported</code>	Platform does not support BT LE
Unauthorized	<code>CBCentralManagerStateUnauthorized</code>	App not authorized to use BT LE
Powered off	<code>CBCentralManagerStatePoweredOff</code>	Bluetooth is powered off
Powered on	<code>CBCentralManagerStatePoweredOn</code>	Bluetooth is powered on and available to use

Check state before using

Make sure BT LE is supported and turned on before using `CBCentralManager`

Demo

Khrob Edmonds
Bluetooth Tools Engineer

More Information

Stephen Chick

iOS Evangelism
chick@apple.com

Craig Keithley

MFi & I/O Technology Evangelist
keithley@apple.com

Developer Programs

MFi Program
<http://developer.apple.com/mfi/>

Bluetooth Mailing List

bluetooth-dev@lists.apple.com

Bluetooth Accessory Design Guidelines

<http://developer.apple.com/hardwaredrivers/>

Reporting Bugs

<http://bugreport.apple.com>

Apple Developer Forums

<http://devforums.apple.com>

Related Sessions

Core Bluetooth 101

Russian Hill
Tuesday 11:30AM

Labs

Bluetooth Lab

Core OS Lab A
Tuesday 3:15PM

Summary

- Enables **your** app to talk to
 - Mobile devices with really small batteries
 - iOS devices
- All while app is backgrounded

 WWDC2012