Core OS

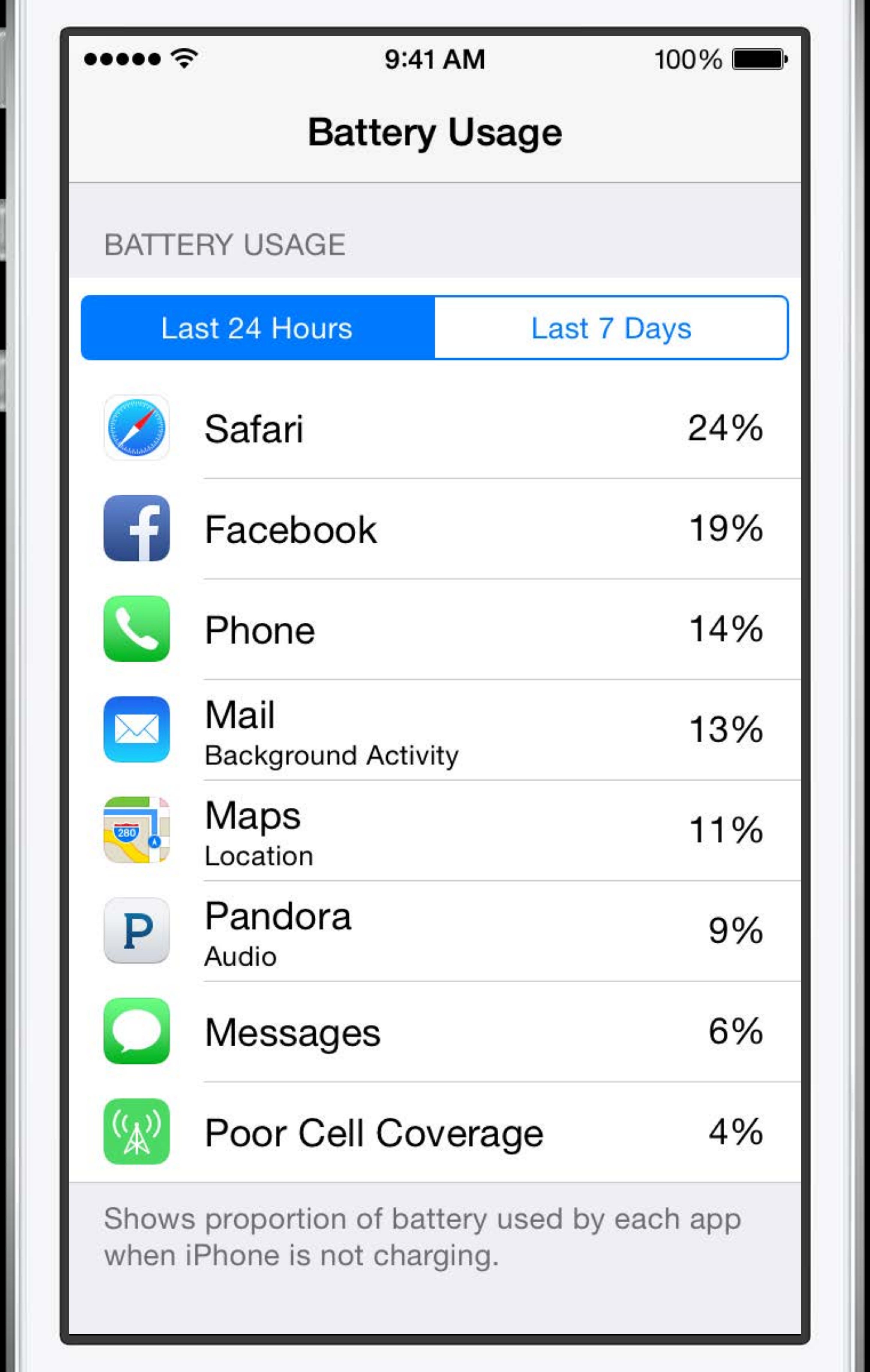
#WWDC14

Writing Energy Efficient Code

Part 2

Session 712

Albert Liu
iOS Software Power Engineer



Battery Usage

BATTERY USAGE

Safari	24%
Facebook	19%
Phone	14%
Mail Background Activity	13%
Maps Location	11%
Pandora Audio	9%
Messages	6%
Poor Cell Coverage	4%

Shows proportion of battery used by each app when iPhone is not charging.

Agenda

Quick Recap

Energy Efficient Networking

Measuring Impact

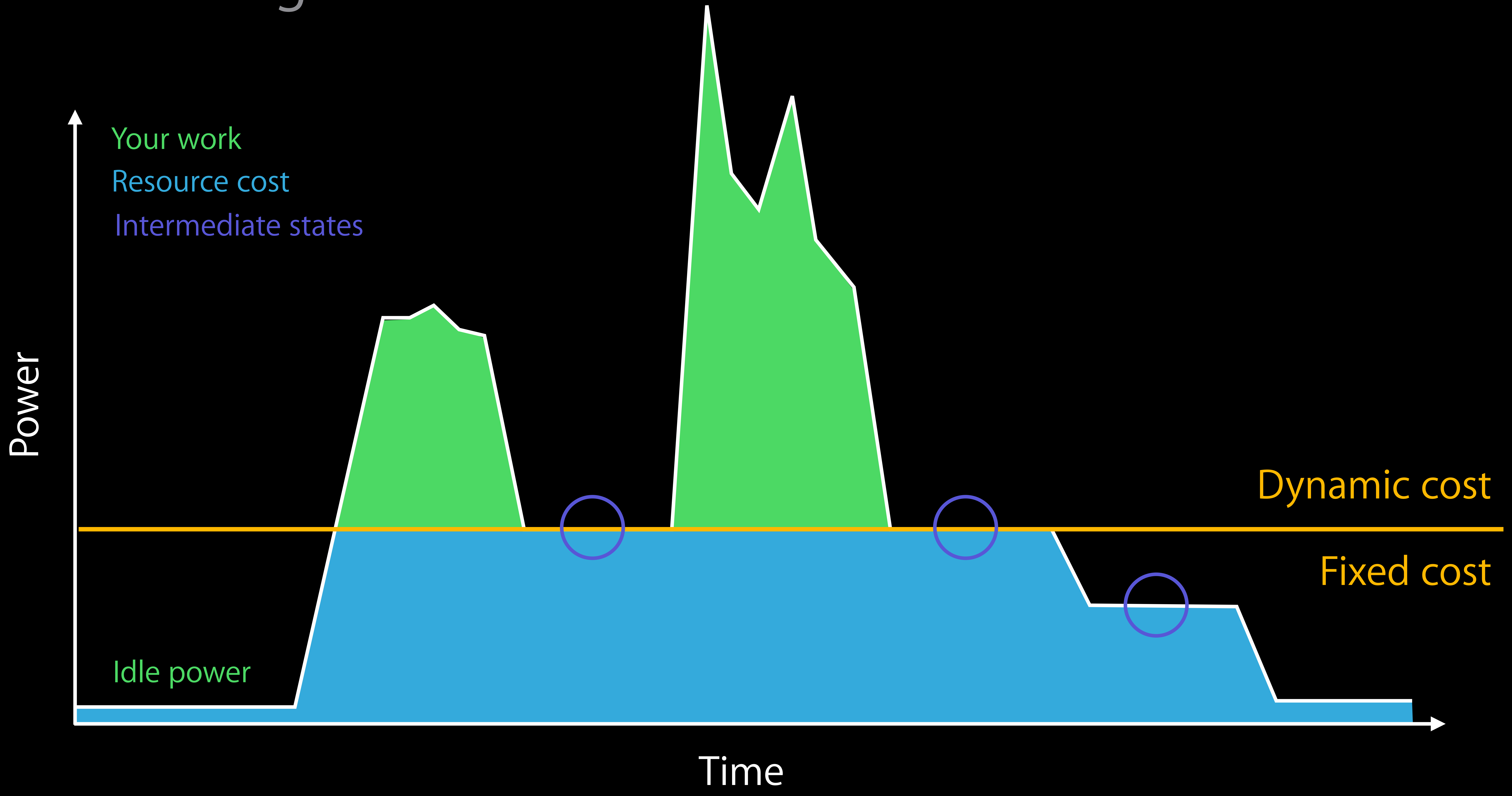
Sleep

Final Thoughts

Quick Recap

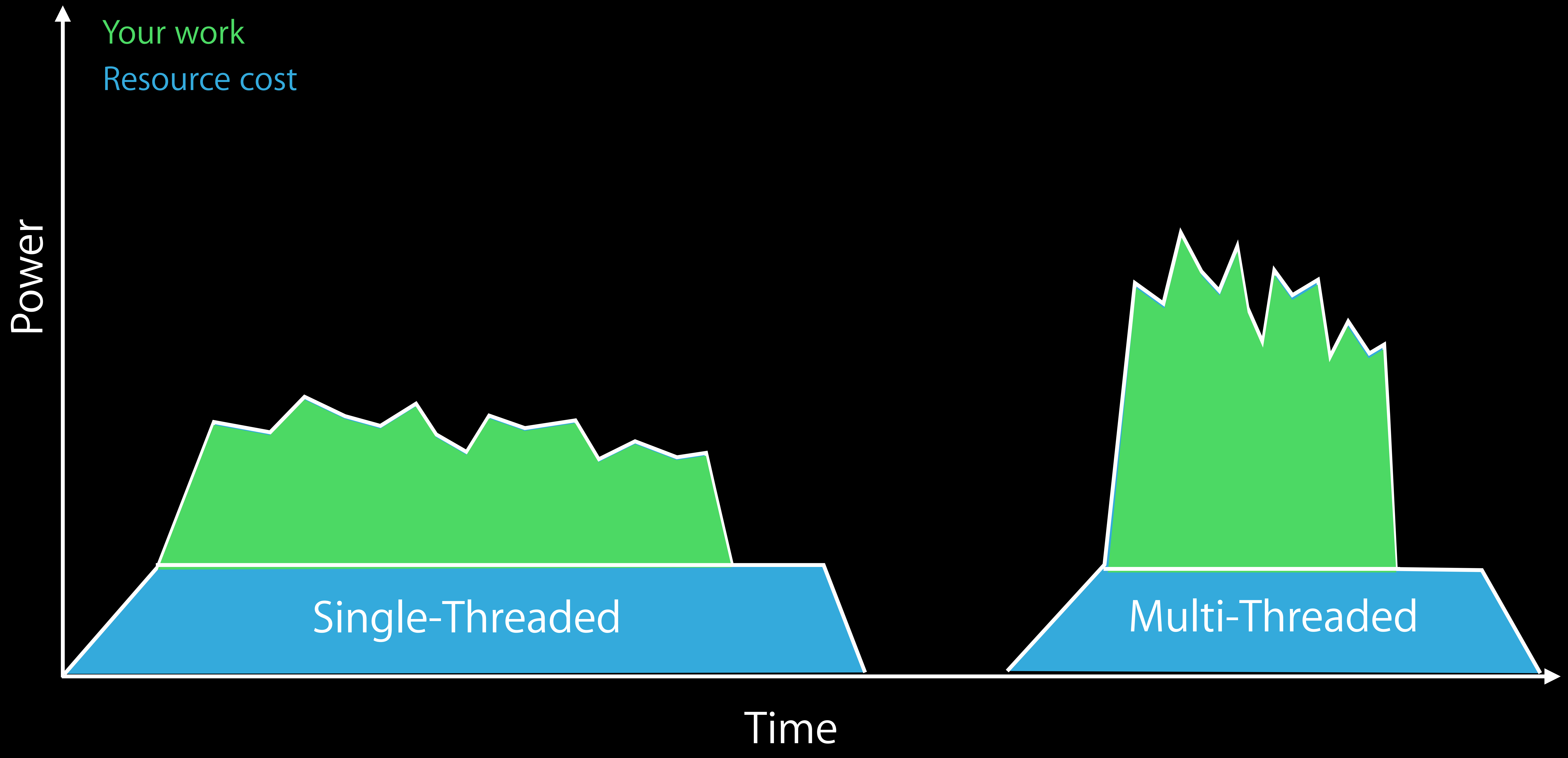
Energy 101

Minimizing overhead cost



Energy 101

Trading power for energy



Reducing Energy Use

What it comes down to…

Do it never

Do it at a better time

Do it more efficiently

Do it less

Energy Efficient Coding

Optimizing CPU usage

Energy Efficient Coding

Optimizing CPU usage

Do it never

Energy Efficient Coding

Optimizing CPU usage

Do it never

- Stop unnecessary work on app transitions

Energy Efficient Coding

Optimizing CPU usage

Do it never

- Stop unnecessary work on app transitions

Do it at a better time

Energy Efficient Coding

Optimizing CPU usage

Do it never

- Stop unnecessary work on app transitions

Do it at a better time

- Scheduling with `NSBackgroundActivityScheduler` / `NSURLSession`

Energy Efficient Coding

Optimizing CPU usage

Do it never

- Stop unnecessary work on app transitions

Do it at a better time

- Scheduling with `NSBackgroundActivityScheduler` / `NSURLSession`

Do it more efficiently

Energy Efficient Coding

Optimizing CPU usage

Do it never

- Stop unnecessary work on app transitions

Do it at a better time

- Scheduling with `NSBackgroundActivityScheduler` / `NSURLSession`

Do it more efficiently

- Set appropriate QoS work priorities

Energy Efficient Coding

Optimizing CPU usage

Do it never

- Stop unnecessary work on app transitions

Do it at a better time

- Scheduling with `NSBackgroundActivityScheduler` / `NSURLSession`

Do it more efficiently

- Set appropriate QoS work priorities

Do it less

Energy Efficient Coding

Optimizing CPU usage

Do it never

- Stop unnecessary work on app transitions

Do it at a better time

- Scheduling with `NSBackgroundActivityScheduler` / `NSURLSession`

Do it more efficiently

- Set appropriate QoS work priorities

Do it less

- Coalesce timers => let CPU idle

CPU Monitor

Runaway background usage

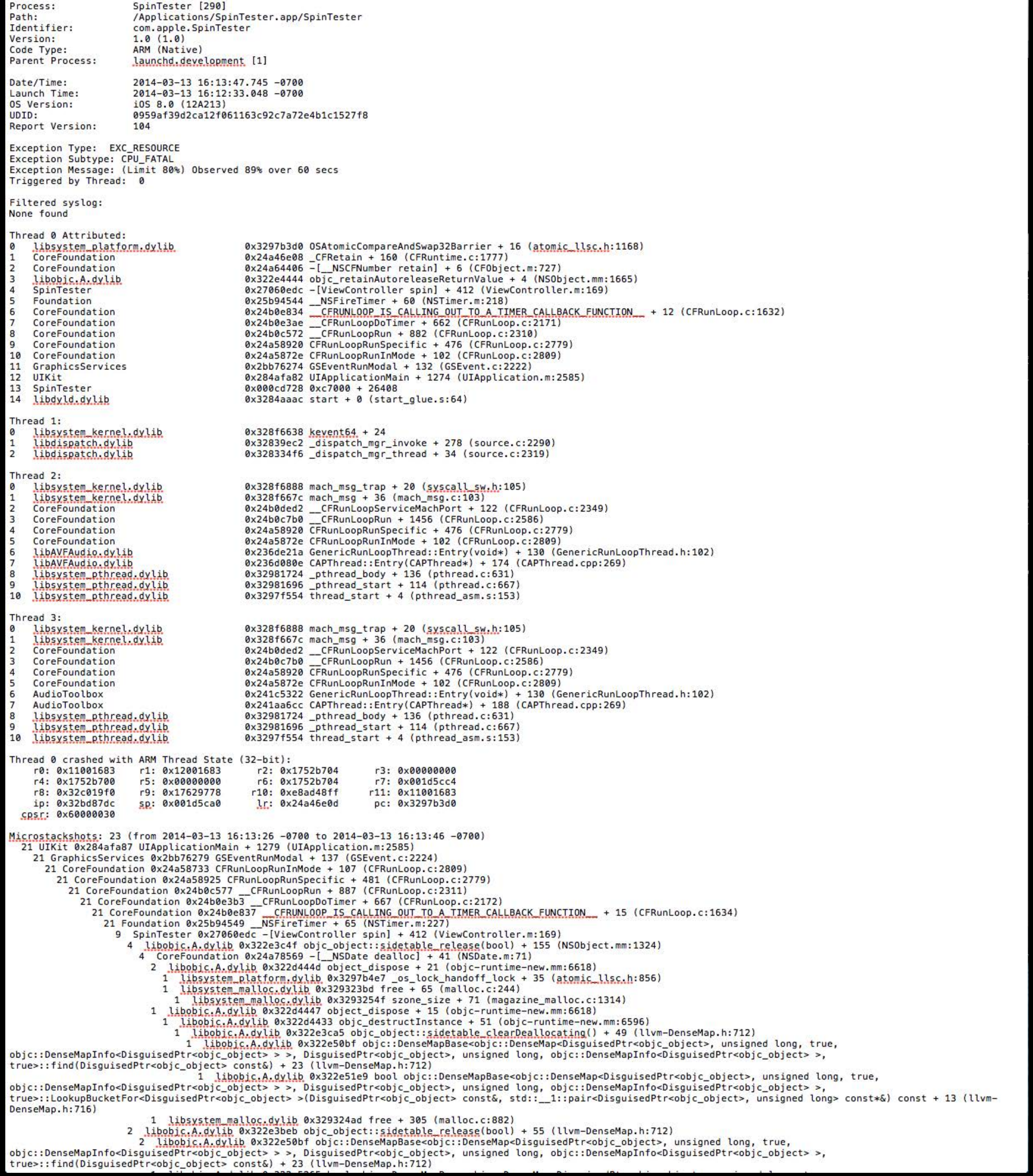
New in iOS 8

Background applications

- Termination when runaway usage is detected
- EXC_RESOURCE : CPU_FATAL Crash log

CPU Monitor

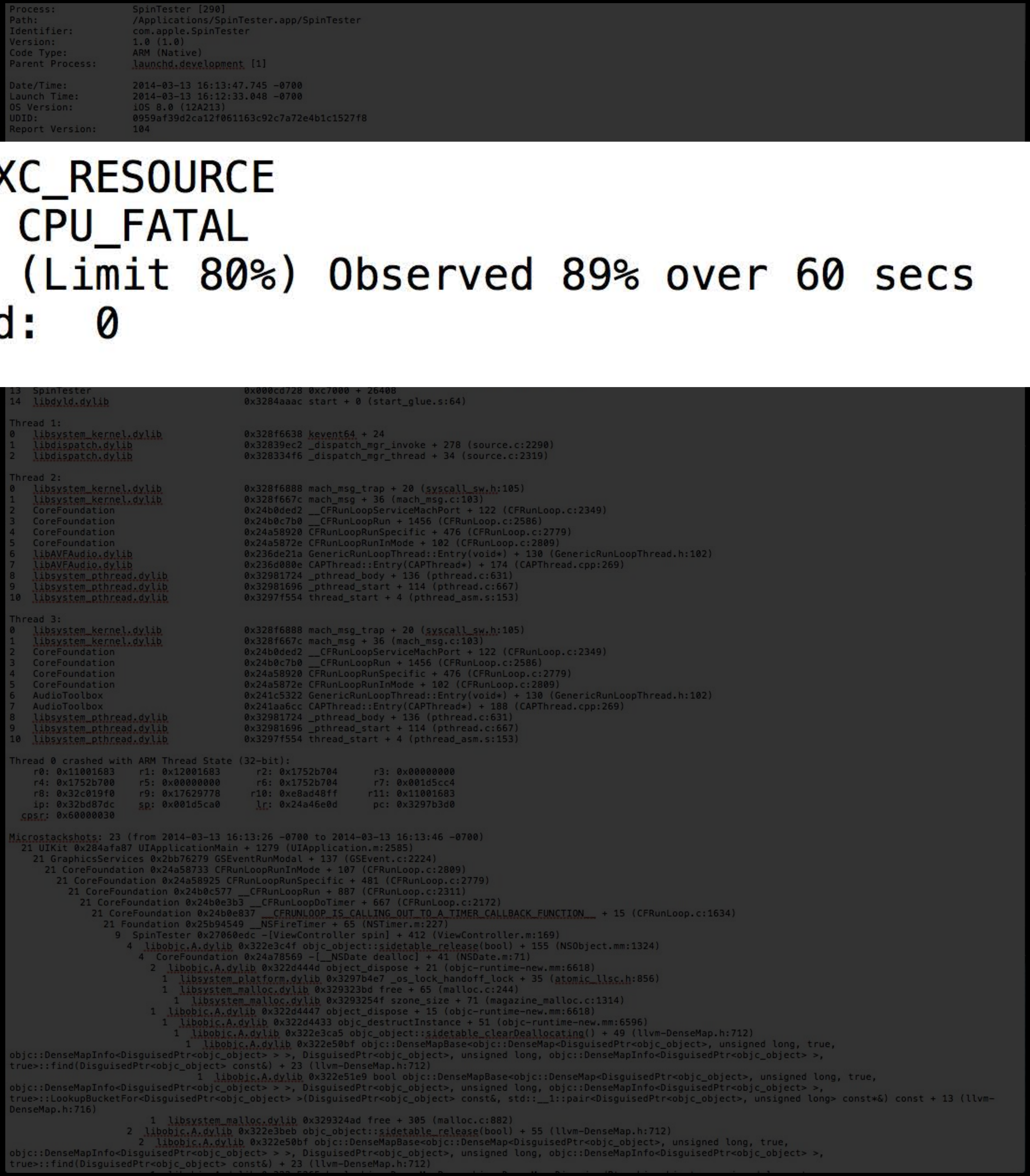
Runaway background usage



```
Process:               SpinTester [290]
Path:                  /Applications/SpinTester.app/SpinTester
Identifier:            com.apple.SpinTester
Version:               1.0 (1.0)
Code Type:             ARM (Native)
Parent Process:        launchd.development [1]

Date/Time:             2014-03-13 16:13:47.745 -0700
Launch Time:           2014-03-13 16:12:33.048 -0700
OS Version:            iOS 8.0 (12A213)
UDID:                  0959af39d2ca12f061163c92c7a72e4b1c1527f8
Report Version:        104

Exception Type:  EXC_RESOURCE
Exception Subtype: CPU_FATAL
Exception Message: (Limit 80%) Observed 89% over 60 secs
Triggered by Thread:  0

Filtered syslog:
None found

Thread 0 Attributed:
0   libsystem_platform.dylib            0x3297b3d0 OSAtomicCompareAndSwap32Barrier + 16 (atomic_llsc.h:1168)
1   CoreFoundation                      0x24a46e08 _CFRetain + 160 (CFRuntime.c:1777)
2   CoreFoundation                      0x24a64406 -[__NSCFNumber retain] + 6 (CFObject.m:727)
3   libobjc.A.dylib                     0x322e4444 objc_retainAutoreleaseReturnValue + 4 (NSObject.mm:1665)
4   SpinTester                          0x27060edc -[ViewController spin] + 412 (ViewController.m:169)
5   Foundation                          0x25b94544 __NSFireTimer + 60 (NSTimer.m:218)
6   CoreFoundation                      0x24b0e834 __CFRUNLOOP_IS_CALLING_OUT_TO_A_TIMER_CALLBACK_FUNCTION__ + 12 (CFRunLoop.c:1632)
7   CoreFoundation                      0x24b0e3ae __CFRunLoopDoTimer + 662 (CFRunLoop.c:2171)
8   CoreFoundation                      0x24b0c572 __CFRunLoopRun + 882 (CFRunLoop.c:2310)
9   CoreFoundation                      0x24a58920 CFRunLoopRunSpecific + 476 (CFRunLoop.c:2779)
10  CoreFoundation                      0x24a5872e CFRunLoopRunInMode + 102 (CFRunLoop.c:2809)
11  GraphicsServices                    0x2bb76274 GSEventRunModal + 132 (GSEvent.c:2222)
12  UIKit                               0x284afa82 UIApplicationMain + 1274 (UIApplication.m:2585)
13  SpinTester                          0x000cd728 0xc7000 + 26408
14  libdyld.dylib                       0x3284aaac start + 0 (start_glue.s:64)

Thread 1:
0   libsystem_kernel.dylib              0x328f6638 kevent64 + 24
1   libdispatch.dylib                   0x32839ec2 _dispatch_mgr_invoke + 278 (source.c:2290)
2   libdispatch.dylib                   0x328334f6 _dispatch_mgr_thread + 34 (source.c:2319)

Thread 2:
0   libsystem_kernel.dylib              0x328f6888 mach_msg_trap + 20 (syscall_sw.h:105)
1   libsystem_kernel.dylib              0x328f667c mach_msg + 36 (mach_msg.c:103)
2   CoreFoundation                      0x24b0ded2 __CFRunLoopServiceMachPort + 122 (CFRunLoop.c:2349)
3   CoreFoundation                      0x24b0c7b0 __CFRunLoopRun + 1456 (CFRunLoop.c:2586)
4   CoreFoundation                      0x24a58920 CFRunLoopRunSpecific + 476 (CFRunLoop.c:2779)
5   CoreFoundation                      0x24a5872e CFRunLoopRunInMode + 102 (CFRunLoop.c:2809)
6   libAVFAudio.dylib                   0x236de21a GenericRunLoopThread::Entry(void*) + 130 (GenericRunLoopThread.h:102)
7   libAVFAudio.dylib                   0x236d080e CAPThread::Entry(CAPThread*) + 174 (CAPThread.cpp:269)
8   libsystem_pthread.dylib             0x32981724 _pthread_body + 136 (pthread.c:631)
9   libsystem_pthread.dylib             0x32981696 _pthread_start + 114 (pthread.c:667)
10  libsystem_pthread.dylib             0x3297f554 thread_start + 4 (pthread_asm.s:153)

Thread 3:
0   libsystem_kernel.dylib              0x328f6888 mach_msg_trap + 20 (syscall_sw.h:105)
1   libsystem_kernel.dylib              0x328f667c mach_msg + 36 (mach_msg.c:103)
2   CoreFoundation                      0x24b0ded2 __CFRunLoopServiceMachPort + 122 (CFRunLoop.c:2349)
3   CoreFoundation                      0x24b0c7b0 __CFRunLoopRun + 1456 (CFRunLoop.c:2586)
4   CoreFoundation                      0x24a58920 CFRunLoopRunSpecific + 476 (CFRunLoop.c:2779)
5   CoreFoundation                      0x24a5872e CFRunLoopRunInMode + 102 (CFRunLoop.c:2809)
6   AudioToolbox                        0x241c5322 GenericRunLoopThread::Entry(void*) + 130 (GenericRunLoopThread.h:102)
7   AudioToolbox                        0x241aa6cc CAPThread::Entry(CAPThread*) + 188 (CAPThread.cpp:269)
8   libsystem_pthread.dylib             0x32981724 _pthread_body + 136 (pthread.c:631)
9   libsystem_pthread.dylib             0x32981696 _pthread_start + 114 (pthread.c:667)
10  libsystem_pthread.dylib             0x3297f554 thread_start + 4 (pthread_asm.s:153)

Thread 0 crashed with ARM Thread State (32-bit):
    r0: 0x11001683    r1: 0x12001683      r2: 0x1752b704      r3: 0x00000000
    r4: 0x1752b700    r5: 0x00000000      r6: 0x1752b704      r7: 0x001d5cc4
    r8: 0x32c019f0    r9: 0x17629778     r10: 0xe8ad48ff     r11: 0x11001683
    ip: 0x32bd87dc    sp: 0x001d5ca0      lr: 0x24a46e0d      pc: 0x3297b3d0
  cpsr: 0x60000030

Microstackshots: 23 (from 2014-03-13 16:13:26 -0700 to 2014-03-13 16:13:46 -0700)
  21 UIKit 0x284afa87 UIApplicationMain + 1279 (UIApplication.m:2585)
    21 GraphicsServices 0x2bb76279 GSEventRunModal + 137 (GSEvent.c:2224)
      21 CoreFoundation 0x24a58733 CFRunLoopRunInMode + 107 (CFRunLoop.c:2809)
        21 CoreFoundation 0x24a58925 CFRunLoopRunSpecific + 481 (CFRunLoop.c:2779)
          21 CoreFoundation 0x24b0c577 __CFRunLoopRun + 887 (CFRunLoop.c:2311)
            21 CoreFoundation 0x24b0e3b3 __CFRunLoopDoTimer + 667 (CFRunLoop.c:2172)
              21 CoreFoundation 0x24b0e837 __CFRUNLOOP_IS_CALLING_OUT_TO_A_TIMER_CALLBACK_FUNCTION__ + 15 (CFRunLoop.c:1634)
                21 Foundation 0x25b94549 __NSFireTimer + 65 (NSTimer.m:227)
                  9  SpinTester 0x27060edc -[ViewController spin] + 412 (ViewController.m:169)
                    4  libobjc.A.dylib 0x322e3c4f objc_object::sidetable_release(bool) + 155 (NSObject.mm:1324)
                      4  CoreFoundation 0x24a78569 -[__NSDate dealloc] + 41 (NSDate.m:71)
                        2  libobjc.A.dylib 0x322d444d object_dispose + 21 (objc-runtime-new.mm:6618)
                          1  libsystem_platform.dylib 0x3297b4e7 _os_lock_handoff_lock + 35 (atomic_llsc.h:856)
                          1  libsystem_malloc.dylib 0x329323bd free + 65 (malloc.c:244)
                            1  libsystem_malloc.dylib 0x3293254f szone_size + 71 (magazine_malloc.c:1314)
                        1  libobjc.A.dylib 0x322d4447 object_dispose + 15 (objc-runtime-new.mm:6618)
                          1  libobjc.A.dylib 0x322d4433 objc_destructInstance + 51 (objc-runtime-new.mm:6596)
                            1  libobjc.A.dylib 0x322e3ca5 objc_object::sidetable_clearDeallocating() + 49 (llvm-DenseMap.h:712)
                              1  libobjc.A.dylib 0x322e50bf objc::DenseMapBase<objc::DenseMap<DisguisedPtr<objc_object>, unsigned long, true,
objc::DenseMapInfo<DisguisedPtr<objc_object> > >, DisguisedPtr<objc_object>, unsigned long, objc::DenseMapInfo<DisguisedPtr<objc_object> >,
true>::find(DisguisedPtr<objc_object> const&) + 23 (llvm-DenseMap.h:712)
                                1  libobjc.A.dylib 0x322e51e9 bool objc::DenseMapBase<objc::DenseMap<DisguisedPtr<objc_object>, unsigned long, true,
objc::DenseMapInfo<DisguisedPtr<objc_object> > >, DisguisedPtr<objc_object>, unsigned long, objc::DenseMapInfo<DisguisedPtr<objc_object> >,
true>::LookupBucketFor<DisguisedPtr<objc_object> >(DisguisedPtr<objc_object> const&, std::__1::pair<DisguisedPtr<objc_object>, unsigned long> const*&) const + 13 (llvm-
DenseMap.h:716)
                        1  libsystem_malloc.dylib 0x329324ad free + 305 (malloc.c:882)
                    2  libobjc.A.dylib 0x322e3beb objc_object::sidetable_release(bool) + 55 (llvm-DenseMap.h:712)
                      2  libobjc.A.dylib 0x322e50bf objc::DenseMapBase<objc::DenseMap<DisguisedPtr<objc_object>, unsigned long, true,
objc::DenseMapInfo<DisguisedPtr<objc_object> > >, DisguisedPtr<objc_object>, unsigned long, objc::DenseMapInfo<DisguisedPtr<objc_object> >,
true>::find(DisguisedPtr<objc_object> const&) + 23 (llvm-DenseMap.h:712)
```

CPU Monitor

Runaway background usage



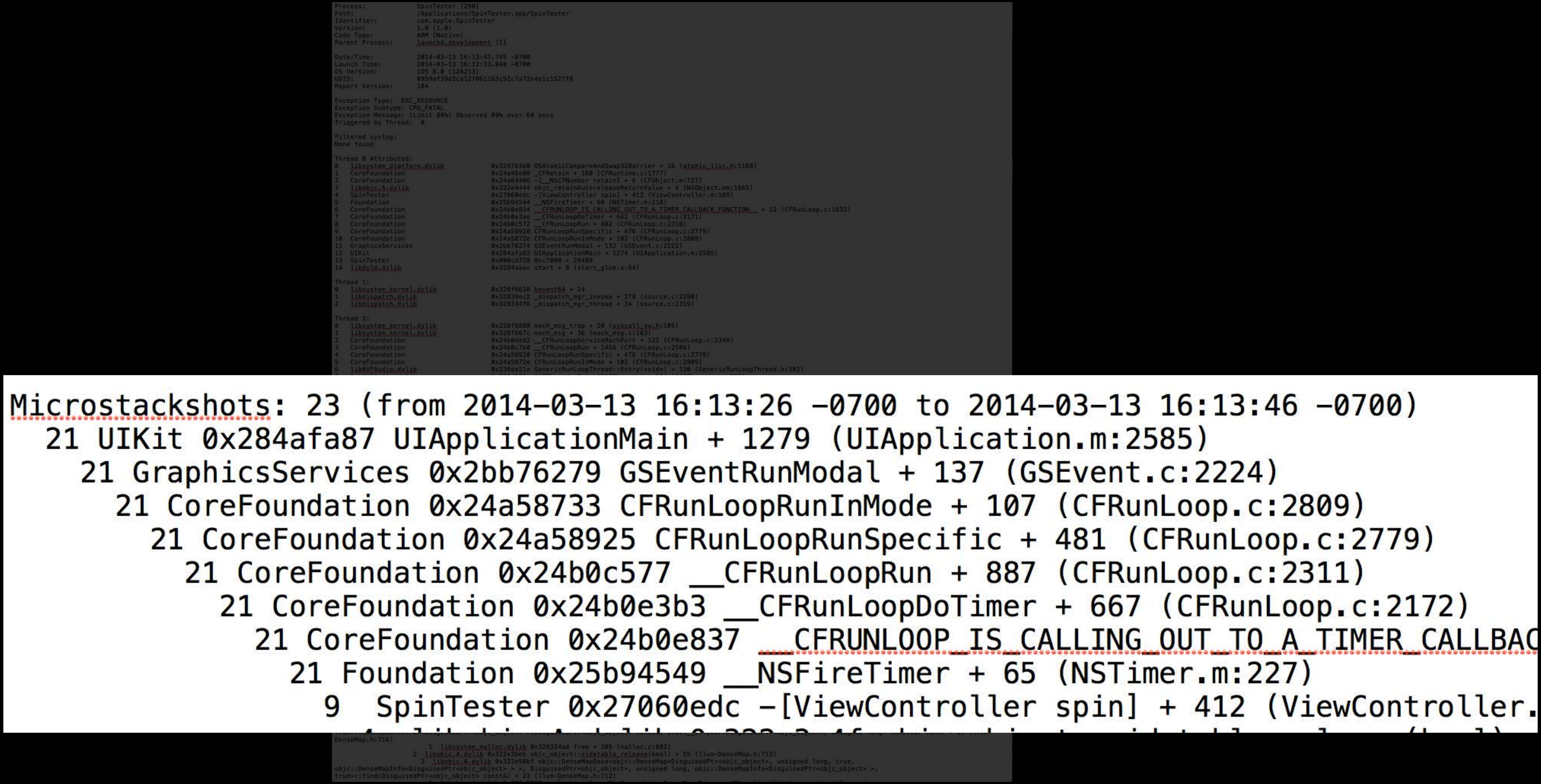
```
Process:               SpinTester [290]
Path:                  /Applications/SpinTester.app/SpinTester
Identifier:            com.apple.SpinTester
Version:               1.0 (1.0)
Code Type:             ARM (Native)
Parent Process:        launchd.development [1]

Date/Time:             2014-03-13 16:13:47.745 -0700
Launch Time:           2014-03-13 16:12:33.048 -0700
OS Version:            iOS 8.0 (12A213)
UDID:                  0959af39d2ca12f061163c92c7a72e4b1c1527f8
Report Version:        104
```

```
Exception Type:  EXC_RESOURCE
Exception Subtype: CPU_FATAL
Exception Message: (Limit 80%) Observed 89% over 60 secs
Triggered by Thread:  0
```

```
13  SpinTester                    0x000cd728 0xc7000 + 26408
14  libdyld.dylib                 0x3284aaac start + 0 (start_glue.s:64)

Thread 1:
0   libsystem_kernel.dylib        0x328f6638 kevent64 + 24
1   libdispatch.dylib             0x32839ec2 _dispatch_mgr_invoke + 278 (source.c:2290)
2   libdispatch.dylib             0x328334f6 _dispatch_mgr_thread + 34 (source.c:2319)

Thread 2:
0   libsystem_kernel.dylib        0x328f6888 mach_msg_trap + 20 (syscall_sw.h:105)
1   libsystem_kernel.dylib        0x328f667c mach_msg + 36 (mach_msg.c:103)
2   CoreFoundation                0x24b0ded2 __CFRunLoopServiceMachPort + 122 (CFRunLoop.c:2349)
3   CoreFoundation                0x24b0c7b0 __CFRunLoopRun + 1456 (CFRunLoop.c:2586)
4   CoreFoundation                0x24a58920 CFRunLoopRunSpecific + 476 (CFRunLoop.c:2779)
5   CoreFoundation                0x24a5872e CFRunLoopRunInMode + 102 (CFRunLoop.c:2809)
6   libAVFAudio.dylib             0x236de21a GenericRunLoopThread::Entry(void*) + 130 (GenericRunLoopThread.h:102)
7   libAVFAudio.dylib             0x236d080e CAPThread::Entry(CAPThread*) + 174 (CAPThread.cpp:269)
8   libsystem_pthread.dylib       0x32981724 _pthread_body + 136 (pthread.c:631)
9   libsystem_pthread.dylib       0x32981696 _pthread_start + 114 (pthread.c:667)
10  libsystem_pthread.dylib       0x3297f554 thread_start + 4 (pthread_asm.s:153)

Thread 3:
0   libsystem_kernel.dylib        0x328f6888 mach_msg_trap + 20 (syscall_sw.h:105)
1   libsystem_kernel.dylib        0x328f667c mach_msg + 36 (mach_msg.c:103)
2   CoreFoundation                0x24b0ded2 __CFRunLoopServiceMachPort + 122 (CFRunLoop.c:2349)
3   CoreFoundation                0x24b0c7b0 __CFRunLoopRun + 1456 (CFRunLoop.c:2586)
4   CoreFoundation                0x24a58920 CFRunLoopRunSpecific + 476 (CFRunLoop.c:2779)
5   CoreFoundation                0x24a5872e CFRunLoopRunInMode + 102 (CFRunLoop.c:2809)
6   AudioToolbox                  0x241c5322 GenericRunLoopThread::Entry(void*) + 130 (GenericRunLoopThread.h:102)
7   AudioToolbox                  0x241aa6cc CAPThread::Entry(CAPThread*) + 188 (CAPThread.cpp:269)
8   libsystem_pthread.dylib       0x32981724 _pthread_body + 136 (pthread.c:631)
9   libsystem_pthread.dylib       0x32981696 _pthread_start + 114 (pthread.c:667)
10  libsystem_pthread.dylib       0x3297f554 thread_start + 4 (pthread_asm.s:153)

Thread 0 crashed with ARM Thread State (32-bit):
    r0: 0x11001683    r1: 0x12001683      r2: 0x1752b704      r3: 0x00000000
    r4: 0x1752b700    r5: 0x00000000      r6: 0x1752b704      r7: 0x001d5cc4
    r8: 0x32c019f0    r9: 0x17629778     r10: 0xe8ad48ff     r11: 0x11001683
    ip: 0x32bd87dc    sp: 0x001d5ca0      lr: 0x24a46e0d      pc: 0x3297b3d0
  cpsr: 0x60000030

Microstackshots: 23 (from 2014-03-13 16:13:26 -0700 to 2014-03-13 16:13:46 -0700)
  21 UIKit 0x284afa87 UIApplicationMain + 1279 (UIApplication.m:2585)
    21 GraphicsServices 0x2bb76279 GSEventRunModal + 137 (GSEvent.c:2224)
      21 CoreFoundation 0x24a58733 CFRunLoopRunInMode + 107 (CFRunLoop.c:2809)
        21 CoreFoundation 0x24a58925 CFRunLoopRunSpecific + 481 (CFRunLoop.c:2779)
          21 CoreFoundation 0x24b0c577 __CFRunLoopRun + 887 (CFRunLoop.c:2311)
            21 CoreFoundation 0x24b0e3b3 __CFRunLoopDoTimer + 667 (CFRunLoop.c:2172)
              21 CoreFoundation 0x24b0e837 __CFRUNLOOP_IS_CALLING_OUT_TO_A_TIMER_CALLBACK_FUNCTION__ + 15 (CFRunLoop.c:1634)
                21 Foundation 0x25b94549 __NSFireTimer + 65 (NSTimer.m:227)
                  9  SpinTester 0x27060edc -[ViewController spin] + 412 (ViewController.m:169)
                    4  libobjc.A.dylib 0x322e3c4f objc_object::sidetable_release(bool) + 155 (NSObject.mm:1324)
                      4  CoreFoundation 0x24a78569 -[__NSDate dealloc] + 41 (NSDate.m:71)
                        2  libobjc.A.dylib 0x322d444d object_dispose + 21 (objc-runtime-new.mm:6618)
                          1  libsystem_platform.dylib 0x3297b4e7 _os_lock_handoff_lock + 35 (atomic_llsc.h:856)
                          1  libsystem_malloc.dylib 0x329323bd free + 65 (malloc.c:244)
                            1  libsystem_malloc.dylib 0x3293254f szone_size + 71 (magazine_malloc.c:1314)
                        1  libobjc.A.dylib 0x322d4447 object_dispose + 15 (objc-runtime-new.mm:6618)
                          1  libobjc.A.dylib 0x322d4433 objc_destructInstance + 51 (objc-runtime-new.mm:6596)
                            1  libobjc.A.dylib 0x322e3ca5 objc_object::sidetable_clearDeallocating() + 49 (llvm-DenseMap.h:712)
                              1  libobjc.A.dylib 0x322e50bf objc::DenseMapBase<objc::DenseMap<DisguisedPtr<objc_object>, unsigned long, true,
objc::DenseMapInfo<DisguisedPtr<objc_object> > >, DisguisedPtr<objc_object>, unsigned long, objc::DenseMapInfo<DisguisedPtr<objc_object> >,
true>::find(DisguisedPtr<objc_object> const&) + 23 (llvm-DenseMap.h:712)
                                1  libobjc.A.dylib 0x322e51e9 bool objc::DenseMapBase<objc::DenseMap<DisguisedPtr<objc_object>, unsigned long, true,
objc::DenseMapInfo<DisguisedPtr<objc_object> > >, DisguisedPtr<objc_object>, unsigned long, objc::DenseMapInfo<DisguisedPtr<objc_object> >,
true>::LookupBucketFor<DisguisedPtr<objc_object> >(DisguisedPtr<objc_object> const&, std::__1::pair<DisguisedPtr<objc_object>, unsigned long> const*&) const + 13 (llvm-
DenseMap.h:716)
                        1  libsystem_malloc.dylib 0x329324ad free + 305 (malloc.c:882)
                    2  libobjc.A.dylib 0x322e3beb objc_object::sidetable_release(bool) + 55 (llvm-DenseMap.h:712)
                      2  libobjc.A.dylib 0x322e50bf objc::DenseMapBase<objc::DenseMap<DisguisedPtr<objc_object>, unsigned long, true,
objc::DenseMapInfo<DisguisedPtr<objc_object> > >, DisguisedPtr<objc_object>, unsigned long, objc::DenseMapInfo<DisguisedPtr<objc_object> >,
true>::find(DisguisedPtr<objc_object> const&) + 23 (llvm-DenseMap.h:712)
```

CPU Monitor

Runaway background usage



```
Process:             SpinTester [290]
Path:                /Applications/SpinTester.app/SpinTester
Identifier:          com.apple.SpinTester
Version:             1.0 (1.0)
Code Type:           ARM (Native)
Parent Process:      launchd.development [1]

Date/Time:           2014-03-13 16:13:47.745 -0700
Launch Time:         2014-03-13 16:12:33.048 -0700
OS Version:          iOS 8.0 (12A213)
UDID:                0959af39d2ca12f061163c92c7a72e4b1c1527f8
Report Version:      104

Exception Type:  EXC_RESOURCE
Exception Subtype: CPU_FATAL
Exception Message: (Limit 80%) Observed 89% over 60 secs
Triggered by Thread:  0

Filtered syslog:
None found

Thread 0 Attributed:
0   libsystem_platform.dylib        0x3297b3d0 OSAtomicCompareAndSwap32Barrier + 16 (atomic_llsc.h:1168)
1   CoreFoundation                  0x24a46e08 _CFRetain + 160 (CFRuntime.c:1777)
2   CoreFoundation                  0x24a64406 -[__NSCFNumber retain] + 6 (CFObject.m:727)
3   libobjc.A.dylib                 0x322e4444 objc_retainAutoreleaseReturnValue + 4 (NSObject.mm:1665)
4   SpinTester                      0x27060edc -[ViewController spin] + 412 (ViewController.m:169)
5   Foundation                      0x25b94544 __NSFireTimer + 60 (NSTimer.m:218)
6   CoreFoundation                  0x24b0e834 __CFRUNLOOP_IS_CALLING_OUT_TO_A_TIMER_CALLBACK_FUNCTION__ + 12 (CFRunLoop.c:1632)
7   CoreFoundation                  0x24b0e3ae __CFRunLoopDoTimer + 662 (CFRunLoop.c:2171)
8   CoreFoundation                  0x24b0c572 __CFRunLoopRun + 882 (CFRunLoop.c:2310)
9   CoreFoundation                  0x24a58920 CFRunLoopRunSpecific + 476 (CFRunLoop.c:2779)
10  CoreFoundation                  0x24a5872e CFRunLoopRunInMode + 102 (CFRunLoop.c:2809)
11  GraphicsServices                0x2bb76274 GSEventRunModal + 132 (GSEvent.c:2222)
12  UIKit                           0x284afa82 UIApplicationMain + 1274 (UIApplication.m:2585)
13  SpinTester                      0x000cd728 0xc7000 + 26408
14  libdyld.dylib                   0x3284aaac start + 0 (start_glue.s:64)

Thread 1:
0   libsystem_kernel.dylib          0x328f6638 kevent64 + 24
1   libdispatch.dylib               0x32839ec2 _dispatch_mgr_invoke + 278 (source.c:2290)
2   libdispatch.dylib               0x328334f6 _dispatch_mgr_thread + 34 (source.c:2319)

Thread 2:
0   libsystem_kernel.dylib          0x328f6888 mach_msg_trap + 20 (syscall_sw.h:105)
1   libsystem_kernel.dylib          0x328f667c mach_msg + 36 (mach_msg.c:103)
2   CoreFoundation                  0x24b0ded2 __CFRunLoopServiceMachPort + 122 (CFRunLoop.c:2349)
3   CoreFoundation                  0x24b0c7b0 __CFRunLoopRun + 1456 (CFRunLoop.c:2586)
4   CoreFoundation                  0x24a58920 CFRunLoopRunSpecific + 476 (CFRunLoop.c:2779)
5   CoreFoundation                  0x24a5872e CFRunLoopRunInMode + 102 (CFRunLoop.c:2809)
6   libAVFAudio.dylib               0x236de21a GenericRunLoopThread::Entry(void*) + 130 (GenericRunLoopThread.h:102)
```

```
Microstackshots: 23 (from 2014-03-13 16:13:26 -0700 to 2014-03-13 16:13:46 -0700)
  21 UIKit 0x284afa87 UIApplicationMain + 1279 (UIApplication.m:2585)
    21 GraphicsServices 0x2bb76279 GSEventRunModal + 137 (GSEvent.c:2224)
      21 CoreFoundation 0x24a58733 CFRunLoopRunInMode + 107 (CFRunLoop.c:2809)
        21 CoreFoundation 0x24a58925 CFRunLoopRunSpecific + 481 (CFRunLoop.c:2779)
          21 CoreFoundation 0x24b0c577 __CFRunLoopRun + 887 (CFRunLoop.c:2311)
            21 CoreFoundation 0x24b0e3b3 __CFRunLoopDoTimer + 667 (CFRunLoop.c:2172)
              21 CoreFoundation 0x24b0e837 __CFRUNLOOP_IS_CALLING_OUT_TO_A_TIMER_CALLBAC
                21 Foundation 0x25b94549 __NSFireTimer + 65 (NSTimer.m:227)
                  9  SpinTester 0x27060edc -[ViewController spin] + 412 (ViewController.
```

```
DenseMap.h:716)
                  1  libsystem_malloc.dylib 0x329324ad free + 305 (malloc.c:882)
                2  libobjc.A.dylib 0x322e3beb objc_object::sidetable_release(bool) + 55 (llvm-DenseMap.h:712)
                  2  libobjc.A.dylib 0x322e50bf objc::DenseMapBase<objc::DenseMap<DisguisedPtr<objc_object>, unsigned long, true,
objc::DenseMapInfo<DisguisedPtr<objc_object> > >, DisguisedPtr<objc_object>, unsigned long, objc::DenseMapInfo<DisguisedPtr<objc_object> >,
true>::find(DisguisedPtr<objc_object> const&) + 23 (llvm-DenseMap.h:712)
```

Energy Efficient Coding

Graphics and Animations

Energy Efficient Coding

Graphics and Animations

Avoid extraneous screen updates

Energy Efficient Coding

Graphics and Animations

Avoid extraneous screen updates

Review blur effect usage

- Reduce frame changes behind blurs

Agenda

Quick Recap

Energy Efficient Networking

Measuring Impact

Sleep

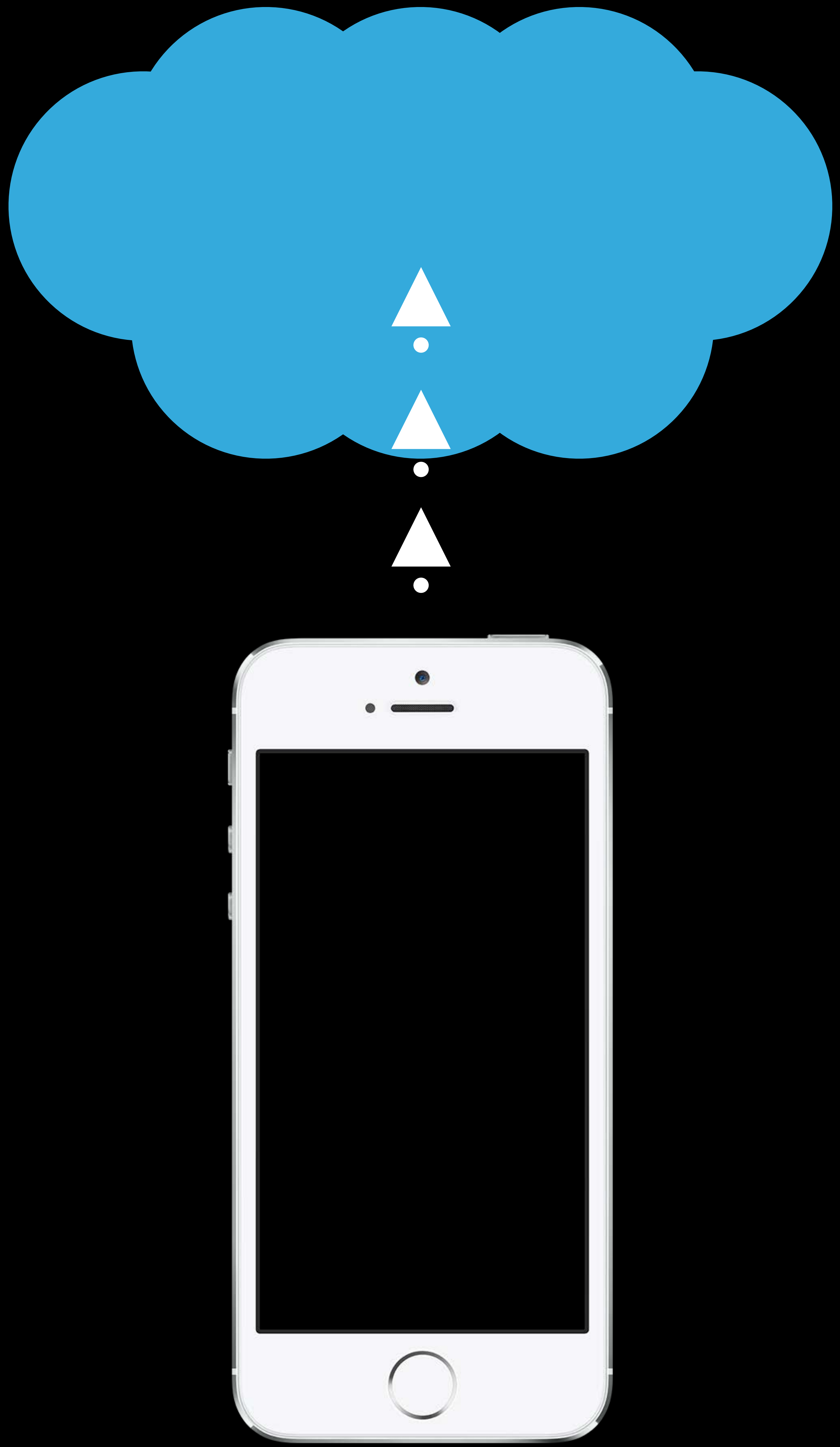
Final Thoughts

Energy Efficient Networking

Web Server Application

Requirements

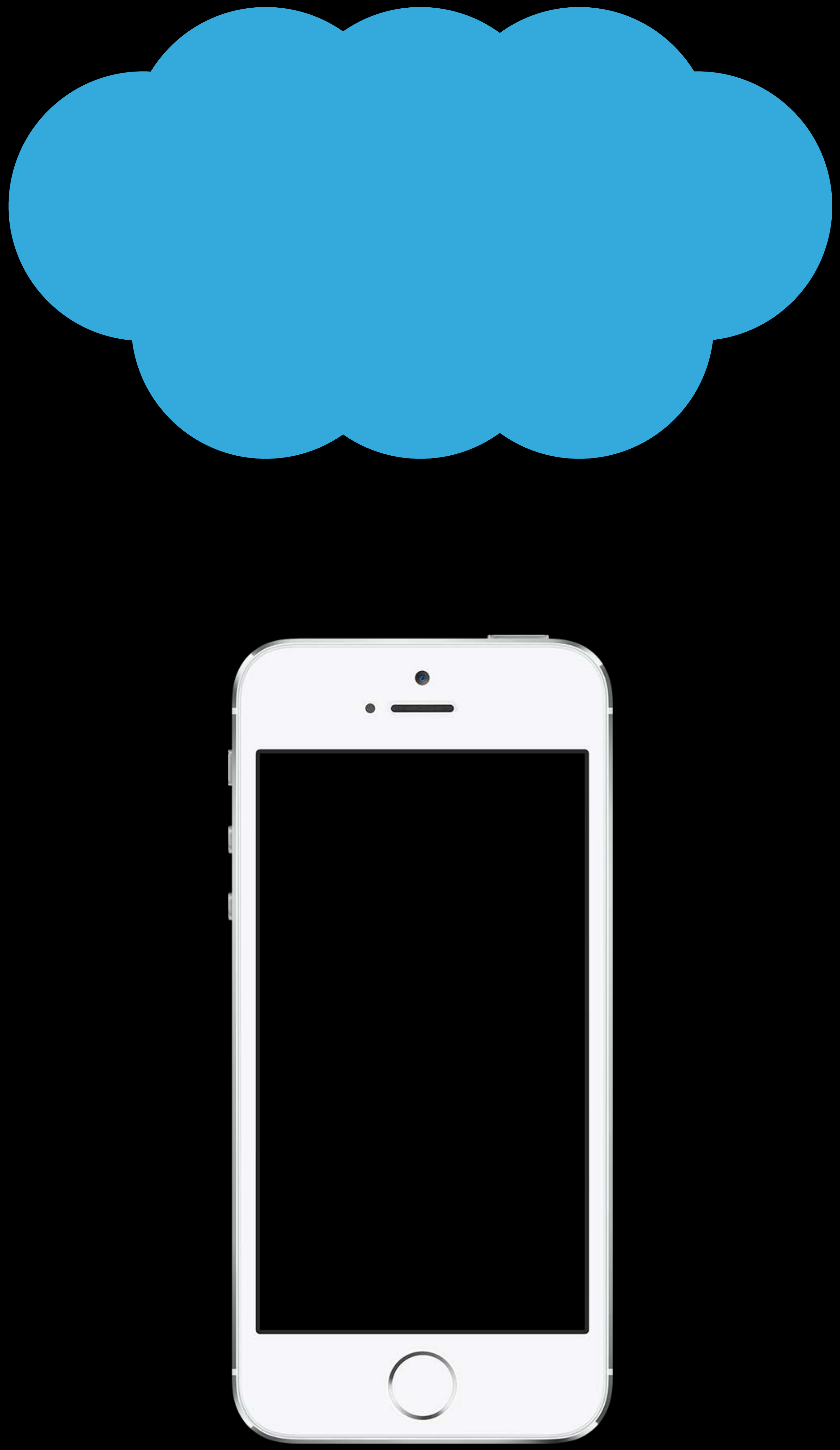
- Sync data to server



Web Server Application

Current solution

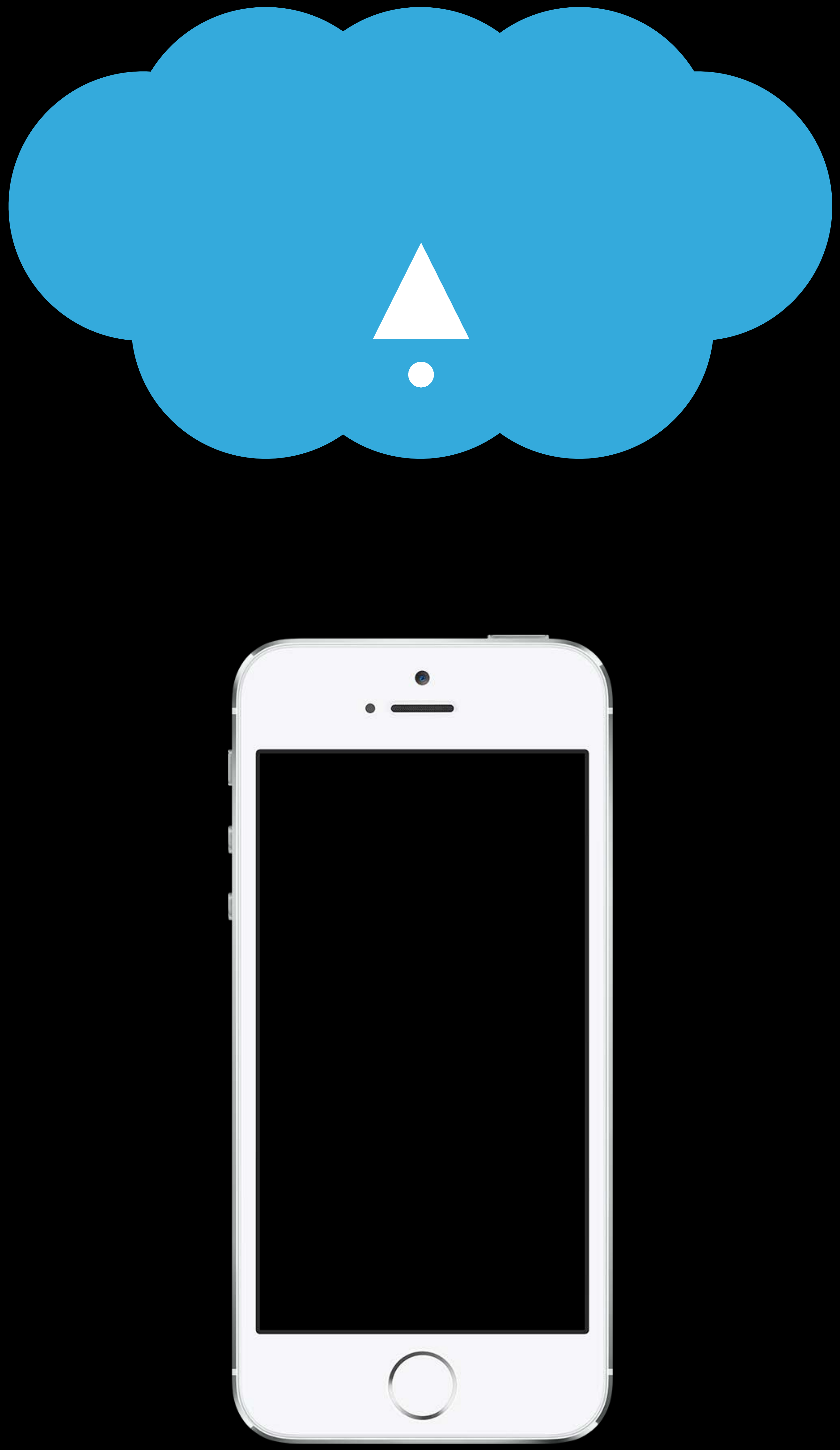
- Sync data as it comes



Web Server Application

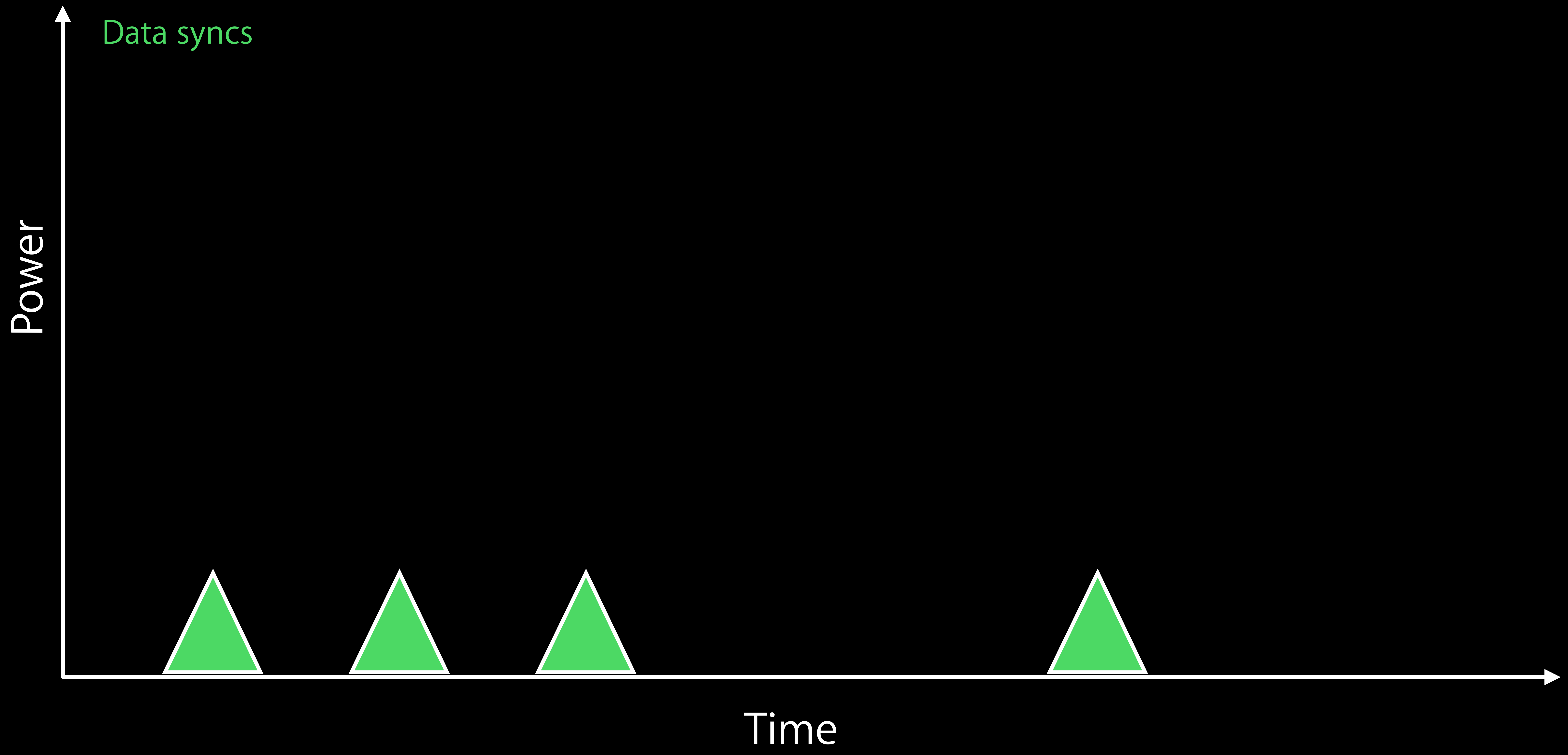
Current solution

- Sync data as it comes



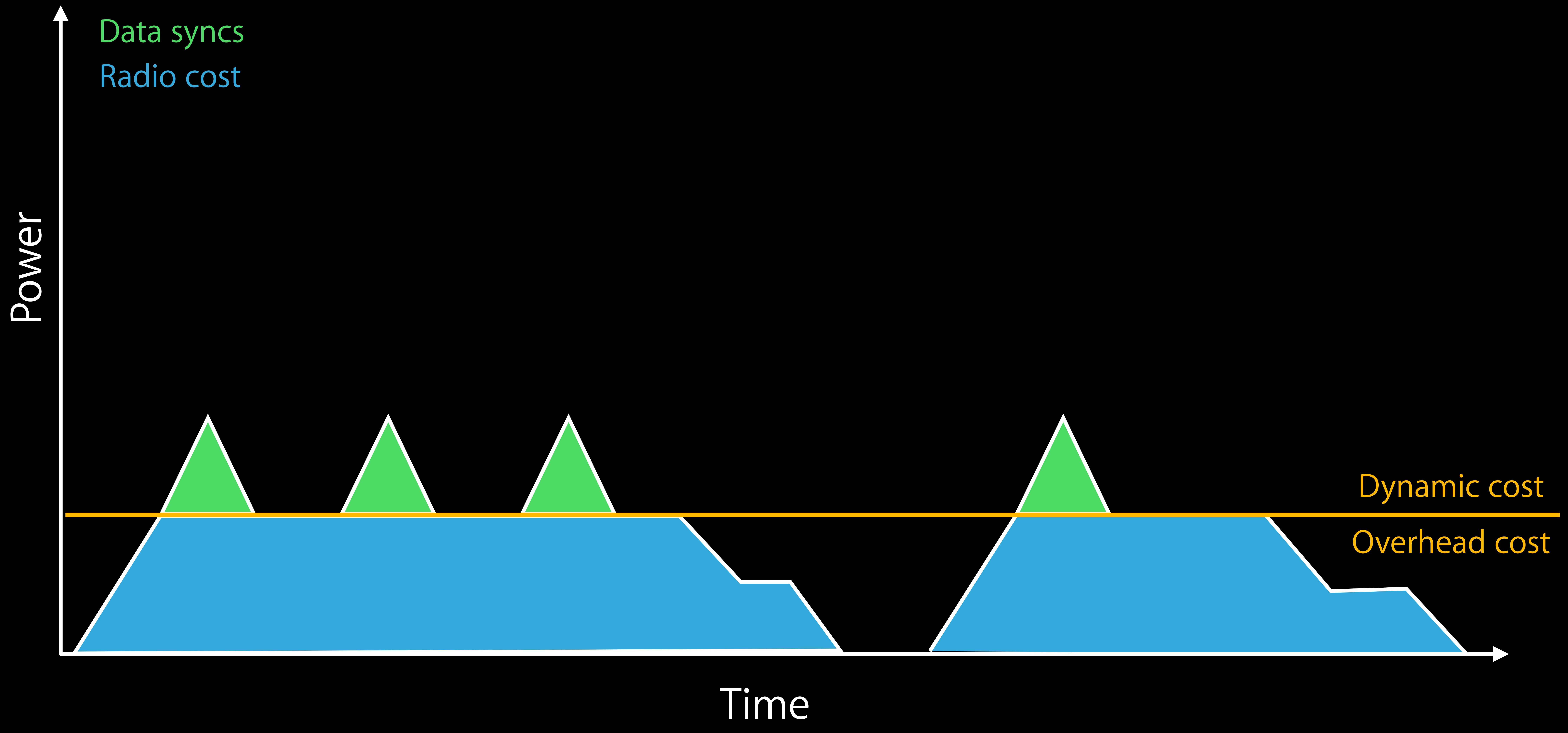
Network Energy

Overhead cost



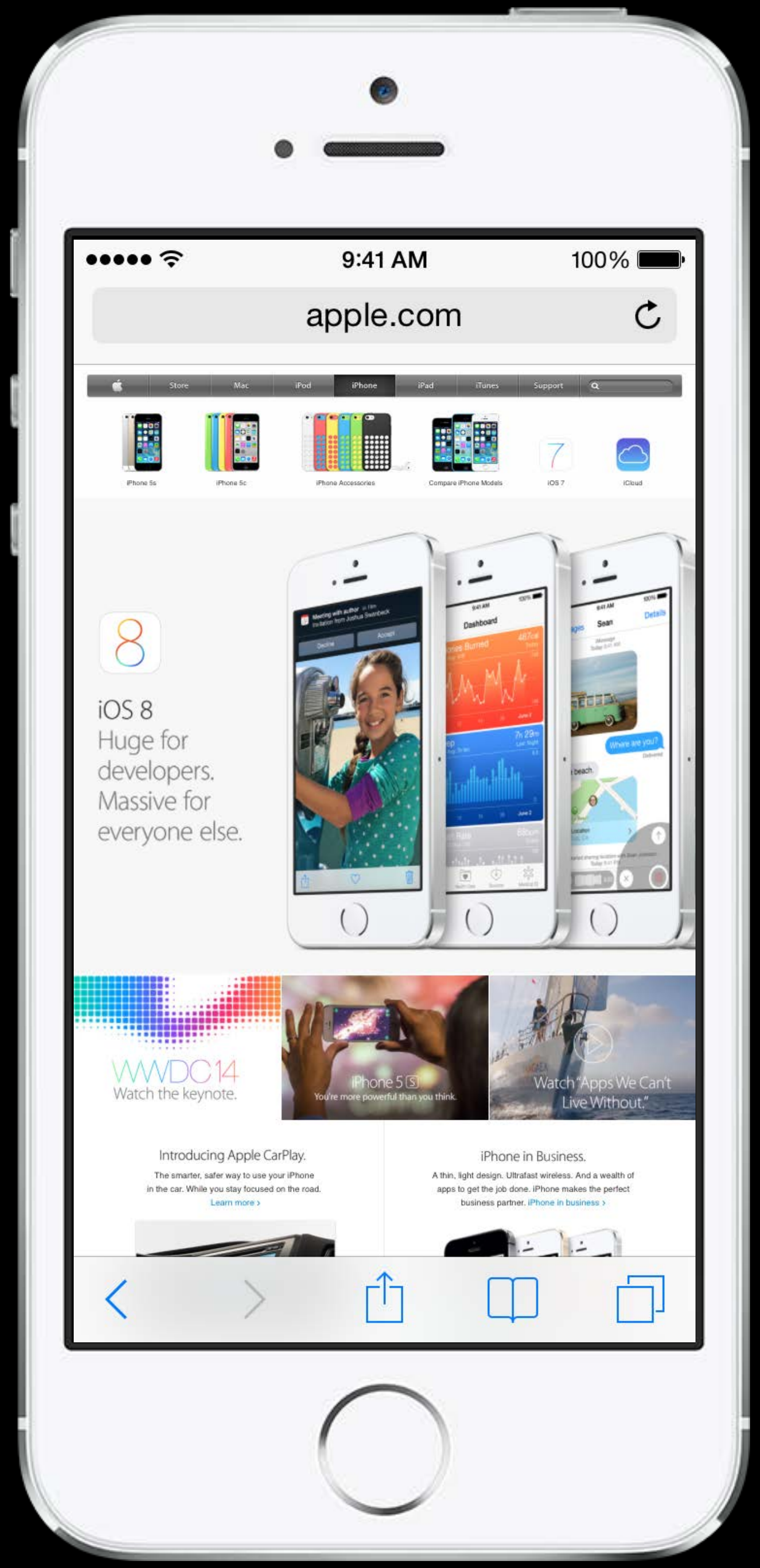
Network Energy

Overhead cost



Network Energy

Web browsing



iOS 8
Huge for developers. Massive for everyone else.

Introducing Apple CarPlay.

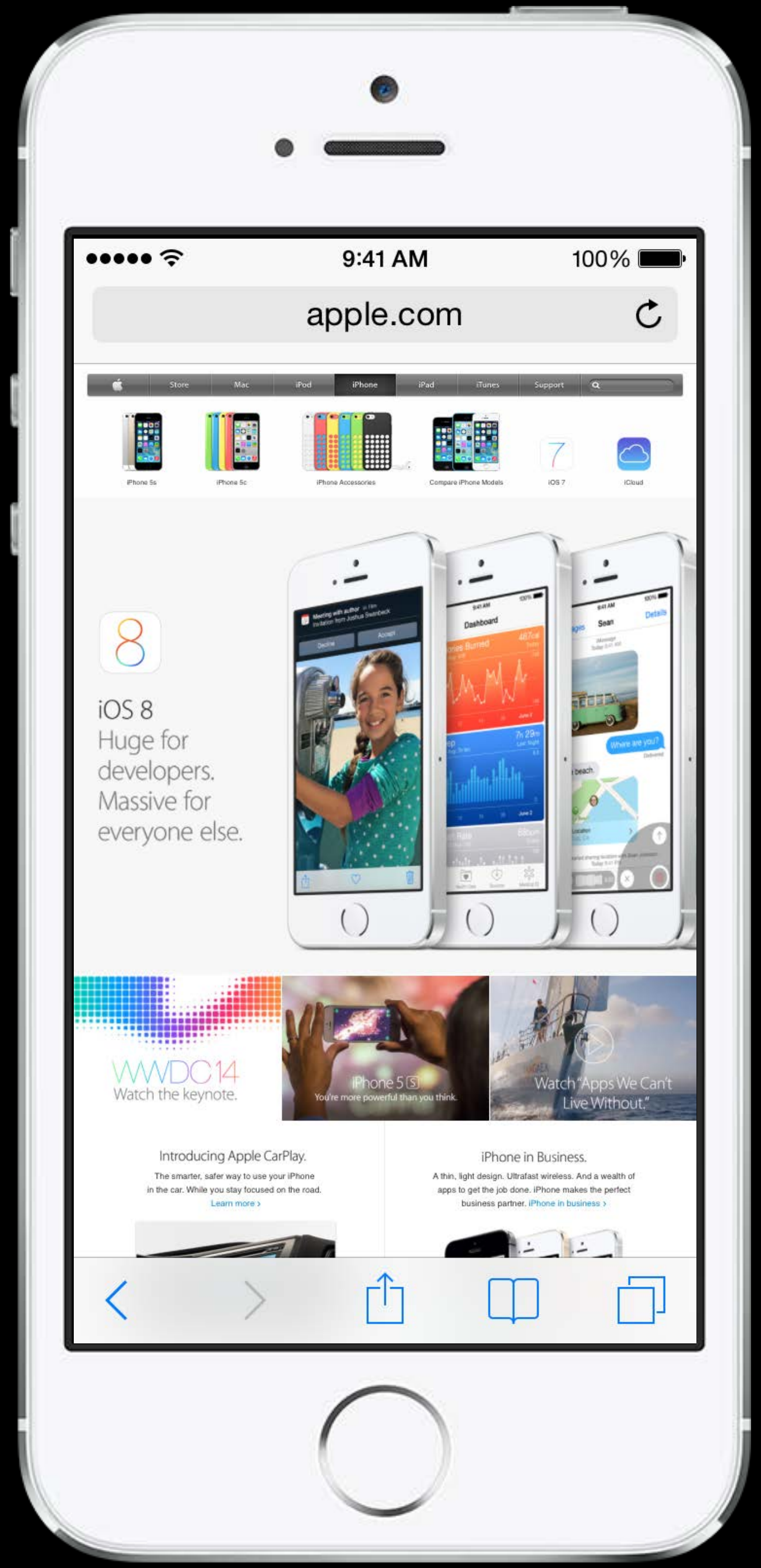
iPhone in Business.

Network Energy

Web browsing

iPhone 5S

- Wifi—10 hours
- 3G—8 hours


9:41 AM
100%
apple.com
iOS 8
Huge for developers. Massive for everyone else.
WWDC14
Watch the keynote.
Watch "Apps We Can't Live Without."
Introducing Apple CarPlay.
iPhone in Business.

Network Energy

All things are NOT created equal

Network Energy

All things are NOT created equal

Cellular vs. Wi-Fi

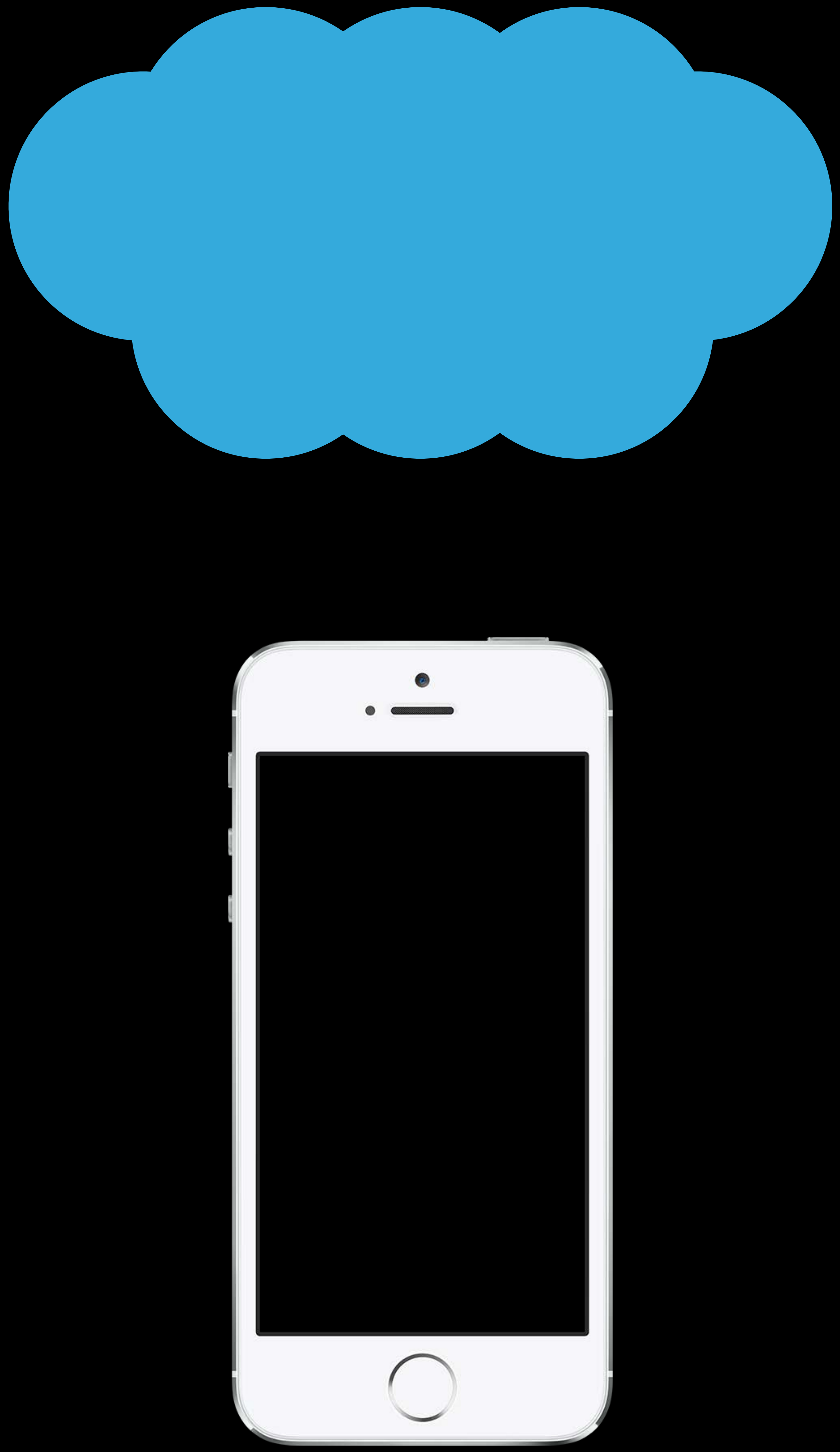
Signal conditions

Network throughput

Webserver Application

Current solution

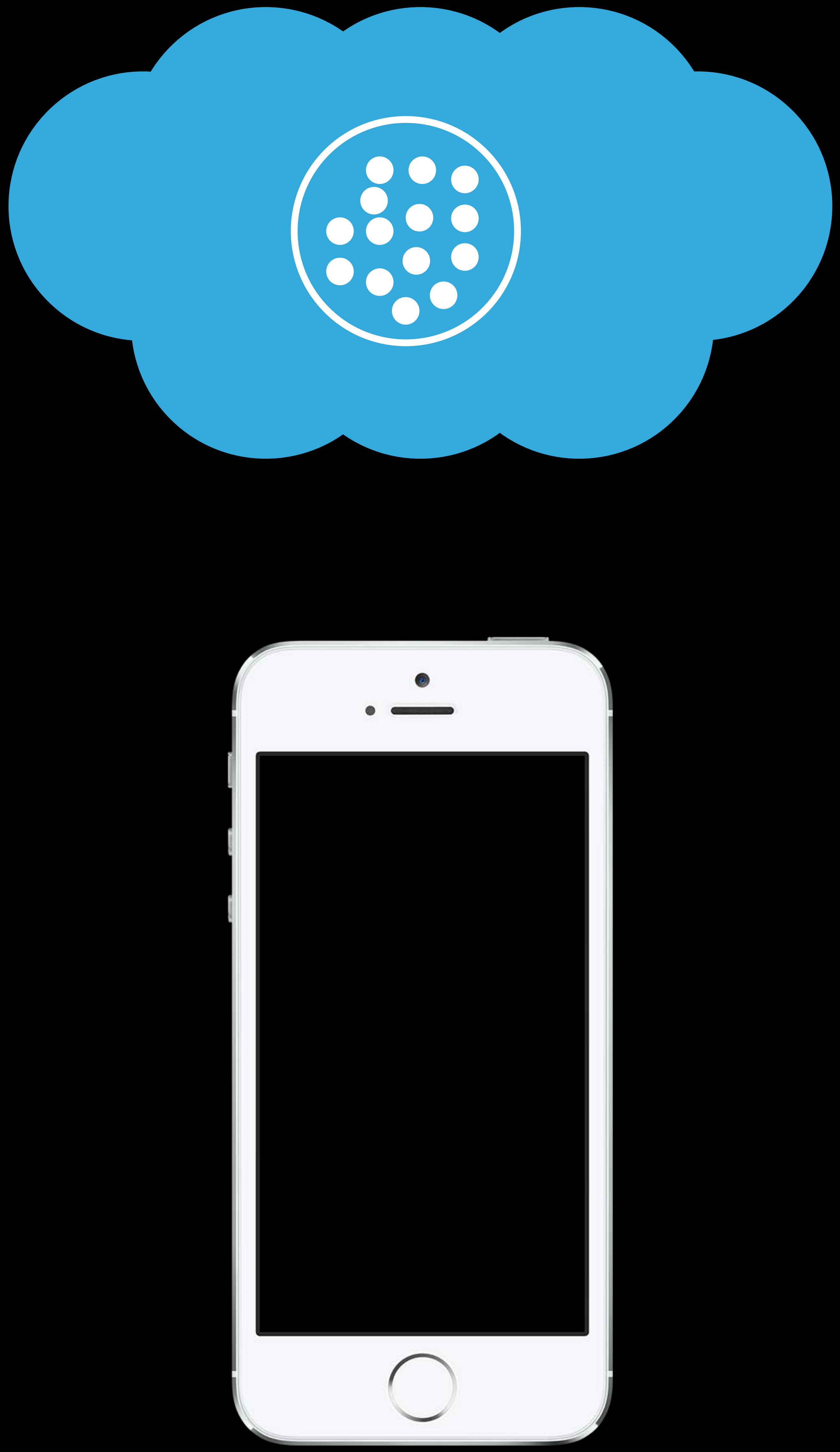
- Buffer data before syncing



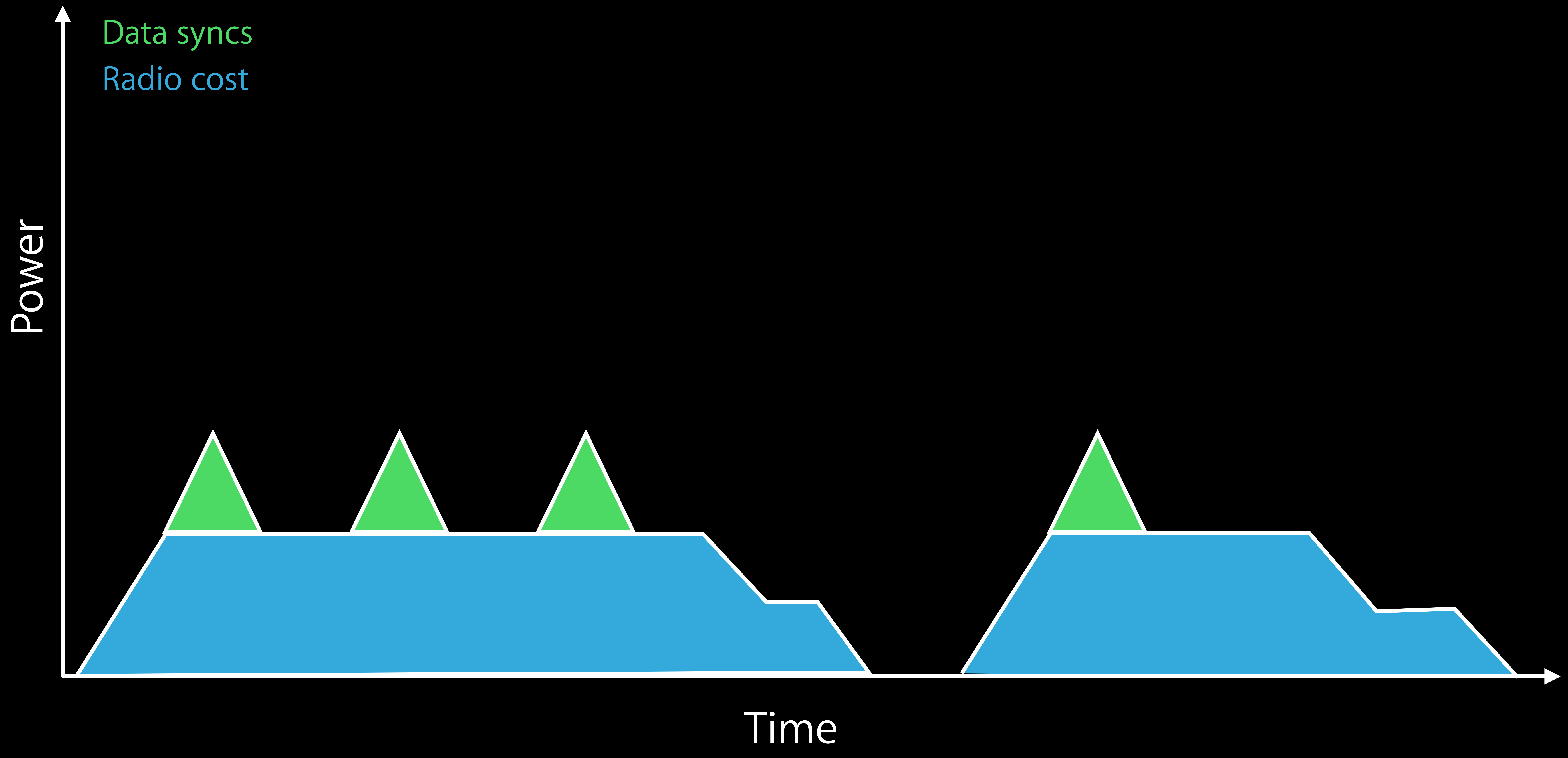
Webserver Application

Current solution

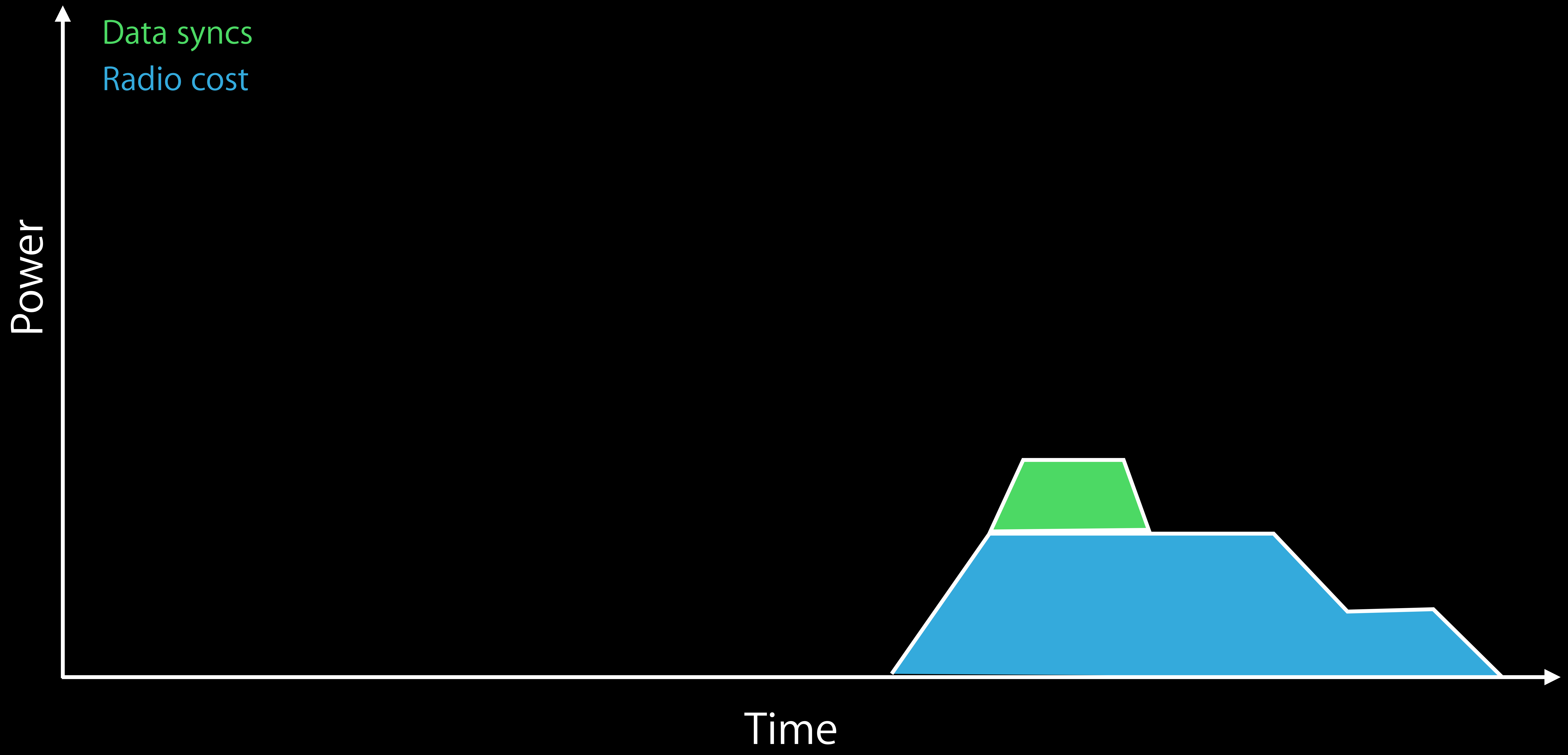
- Buffer data before syncing



Network Energy



Network Energy



Energy Efficient Networking

Concepts to remember

Energy Efficient Networking

Concepts to remember

Doing it more efficiently

- Coalesce Transactions

Energy Efficient Networking

Concepts to remember

Doing it more efficiently

- Coalesce Transactions

Doing it less/never

- Cut down Transfers

Cut Down/Transfers

Doing it never/less

Cut Down/Transfers

Doing it never/less

Reduce data sizes

- Reduce media quality
- Compress your data

Cut Down/Transfers

Doing it never/less

Reduce data sizes

- Reduce media quality
- Compress your data

Avoid redundant transfers

- Caching
- Resumable transactions

Cut Down/Transfers

Doing it never/less

Reduce data sizes

- Reduce media quality
- Compress your data

Avoid redundant transfers

- Caching
- Resumable transactions

Handle errors

- Timeout
- Retry policies

Energy Efficient Networking

Concepts to remember

Doing it more efficiently

- Coalesce Transactions

Doing it less/never

- Cut down Transfers

Doing it at a better time

- Consider Tolerance

Consider Tolerance

Doing it at a better time

Understand requirements

- When is it needed?

Consider

- Technology used
- Network conditions

Webserver Application

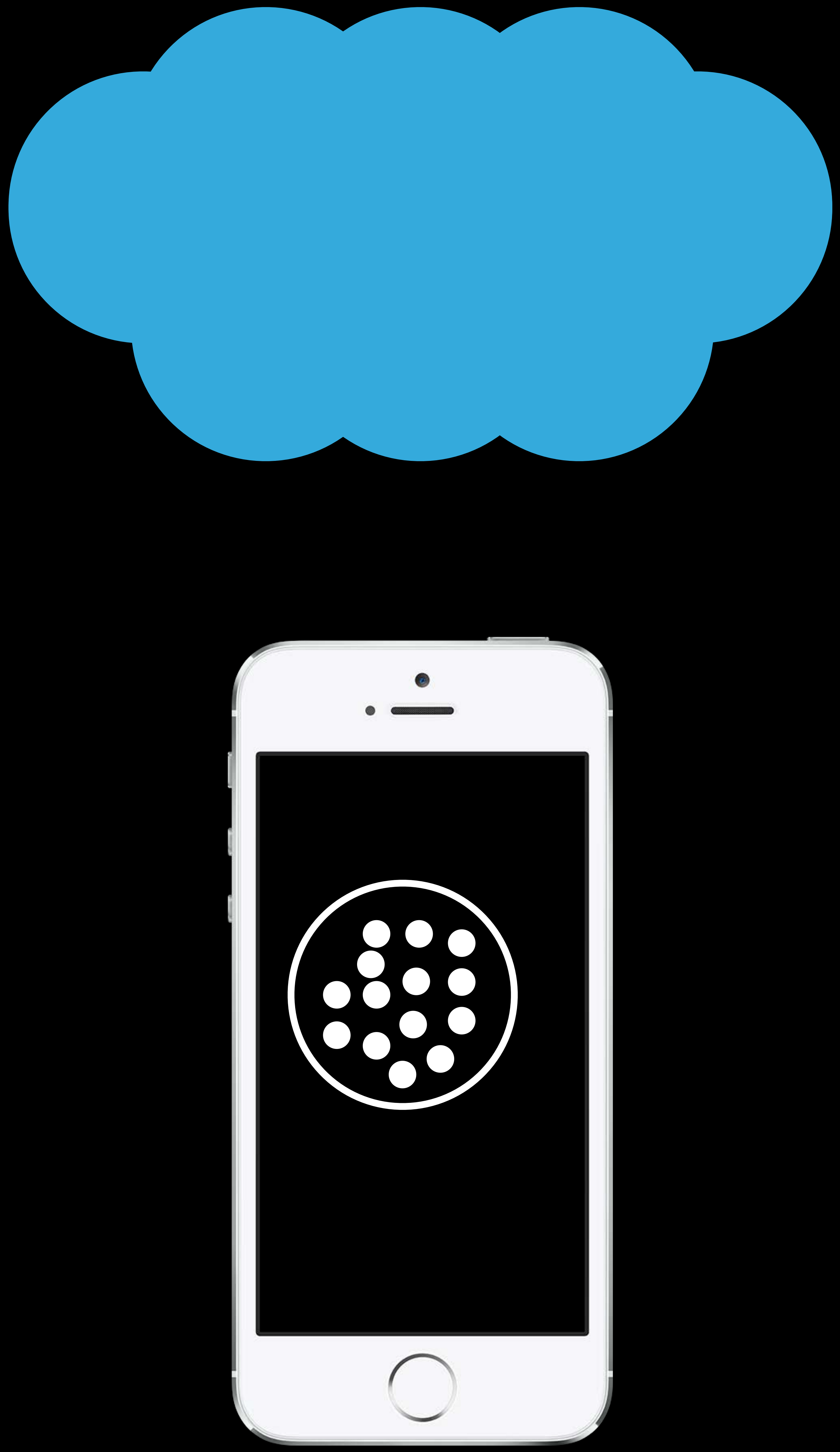
Current solution

- Buffer data before syncing

Webserver Application

Current solution

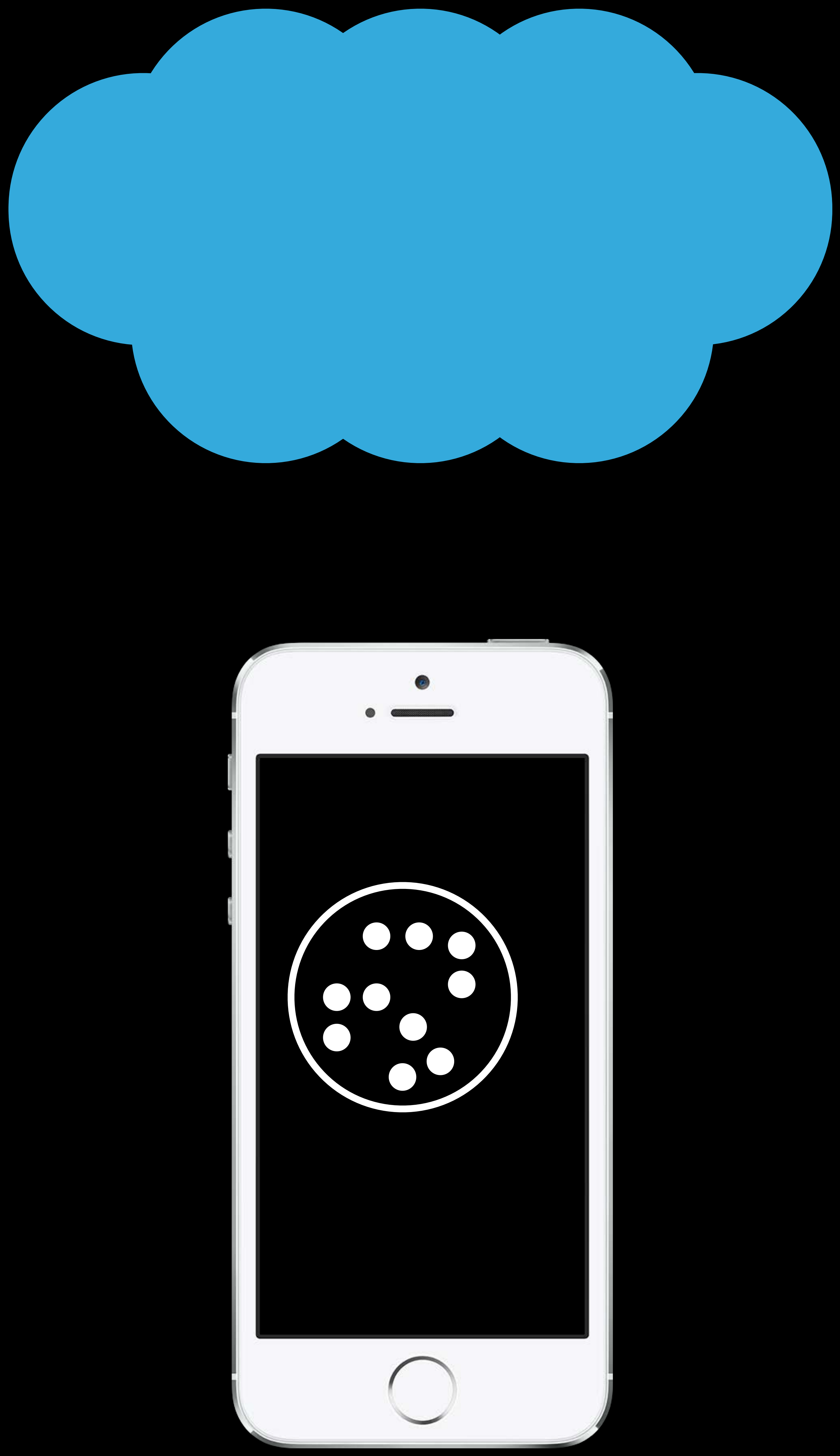
- Buffer data before syncing
- Check data before buffer



Webserver Application

Current solution

- Buffer data before syncing
- Check data before buffer



Webserver Application

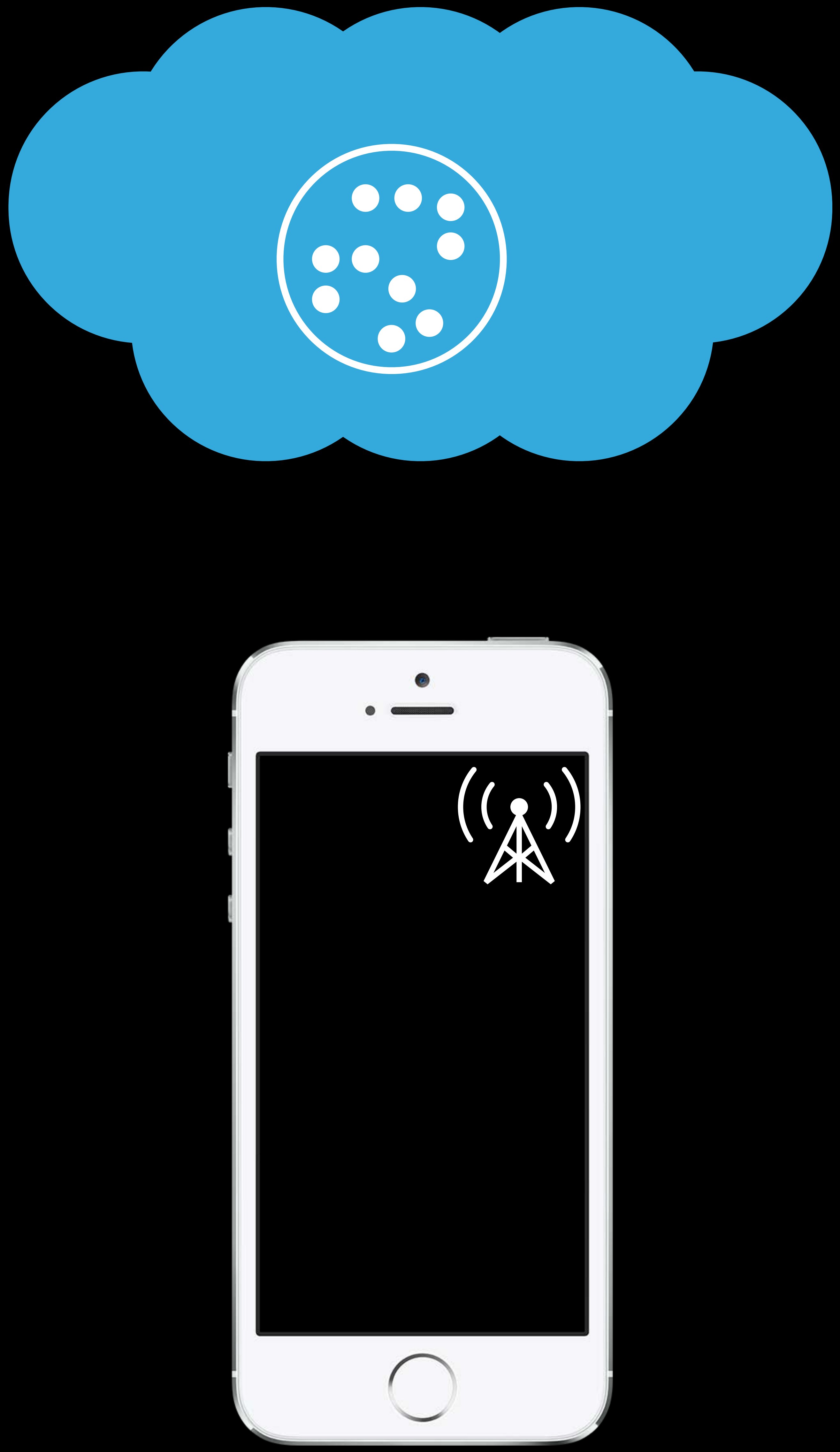
Current solution

- Buffer data before syncing
- Check data before buffer
- Sync on good signal conditions

Webserver Application

Current solution

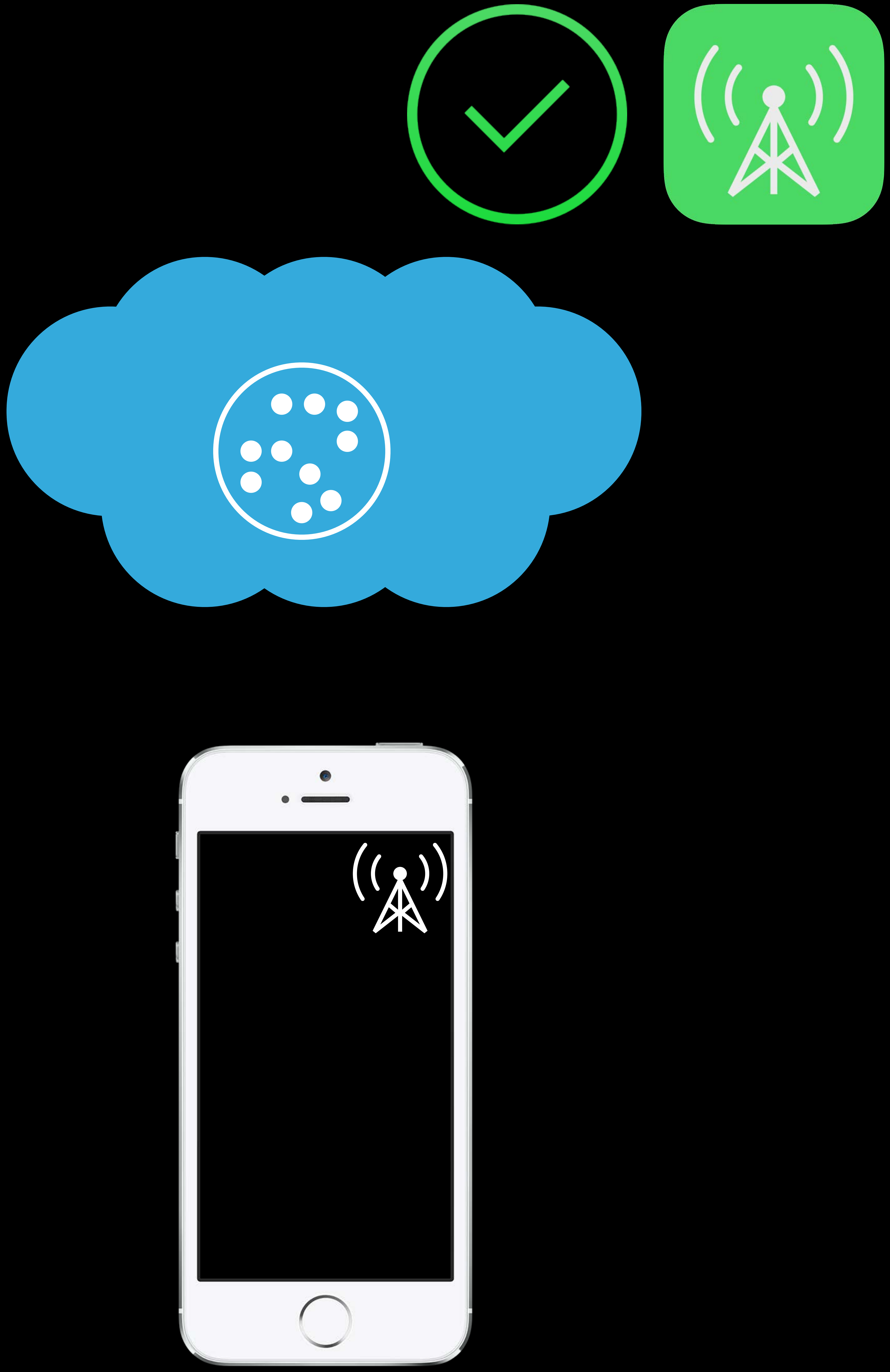
- Buffer data before syncing
- Check data before buffer
- Sync on good signal conditions



Webserver Application

Current solution

- Buffer data before syncing
- Check data before buffer
- Sync on good signal conditions



NSURLSession

Networking API

NSURLSession

Networking API

Pause/Resume

NSURLSession

Networking API

Pause/Resume

Cache per session

NSURLSession

Networking API

Pause/Resume

Cache per session

Background sessions

NSURLSession

Networking API

Pause/Resume

Cache per session

Background sessions

- Out of process transactions

NSURLSession

Networking API

Pause/Resume

Cache per session

Background sessions

- Out of process transactions
- Throughput monitoring

NSURLSession

Networking API

Pause/Resume

Cache per session

Background sessions

- Out of process transactions
- Throughput monitoring
- Automatic retries

NSURLSession

Discretionary tasks

NSURLSession

Discretionary tasks

Available for background sessions

NSURLSession

Discretionary tasks

Available for background sessions

Power optimal system scheduling

NSURLSession

Discretionary tasks

Available for background sessions

Power optimal system scheduling

Adjustable scheduling window

NSURLSession

Discretionary tasks

Available for background sessions

Power optimal system scheduling

Adjustable scheduling window

- `configuration.timeoutIntervalForResource  // should be > 12 hours`

NSURLSession

Discretionary tasks

Available for background sessions

Power optimal system scheduling

Adjustable scheduling window

- `configuration.timeoutIntervalForResource  // should be > 12 hours`
- Error will be thrown if conditions are not met

NSURLSession

```
NSURLSessionConfiguration *config = [NSURLSessionConfiguration
backgroundSessionConfigurationWithIdentifier:@“com.apple.App.UserRestore”];
[config setAllowsCellularAccess:NO];
[config setDiscretionary:YES];
config.timeoutIntervalForResource = 18 * 60 * 60;

NSURLSession *session = [NSURLSession sessionWithConfiguration:config
delegate:self delegateQueue:nil];

NSURL *url = [NSURL URLWithString:@“http://www.foo.com/interesting_thing”];
NSURLSessionTask *task = [session downloadTaskWithURL:url];
[task resume];
```

NSURLSession

```
NSURLSessionConfiguration *config = [NSURLSessionConfiguration
backgroundSessionConfigurationWithIdentifier:@“com.apple.App.UserRestore”];
[config setAllowsCellularAccess:NO];
[config setDiscretionary:YES];
config.timeoutIntervalForResource = 18 * 60 * 60;
```

```
NSURLSession *session = [NSURLSession sessionWithConfiguration:config
delegate:self delegateQueue:nil];

NSURL *url = [NSURL URLWithString:@“http://www.foo.com/interesting_thing”];
NSURLSessionTask *task = [session downloadTaskWithURL:url];
[task resume];
```

NSURLSession

```
NSURLSessionConfiguration *config = [NSURLSessionConfiguration
backgroundSessionConfigurationWithIdentifier:@“com.apple.App.UserRestore”];
[config setAllowsCellularAccess:NO];
[config setDiscretionary:YES];
config.timeoutIntervalForResource = 18 * 60 * 60;
```

```
NSURLSession *session = [NSURLSession sessionWithConfiguration:config
delegate:self delegateQueue:nil];

NSURL *url = [NSURL URLWithString:@“http://www.foo.com/interesting_thing”];
NSURLSessionTask *task = [session downloadTaskWithURL:url];
[task resume];
```

Energy Efficient Networking

Concepts to remember

Doing it more efficiently

- Coalesce Transactions

Doing it less/never

- Cut down Transfers

Doing it at a better time

- Consider Tolerance

Agenda

Quick Recap

Energy Efficient Networking

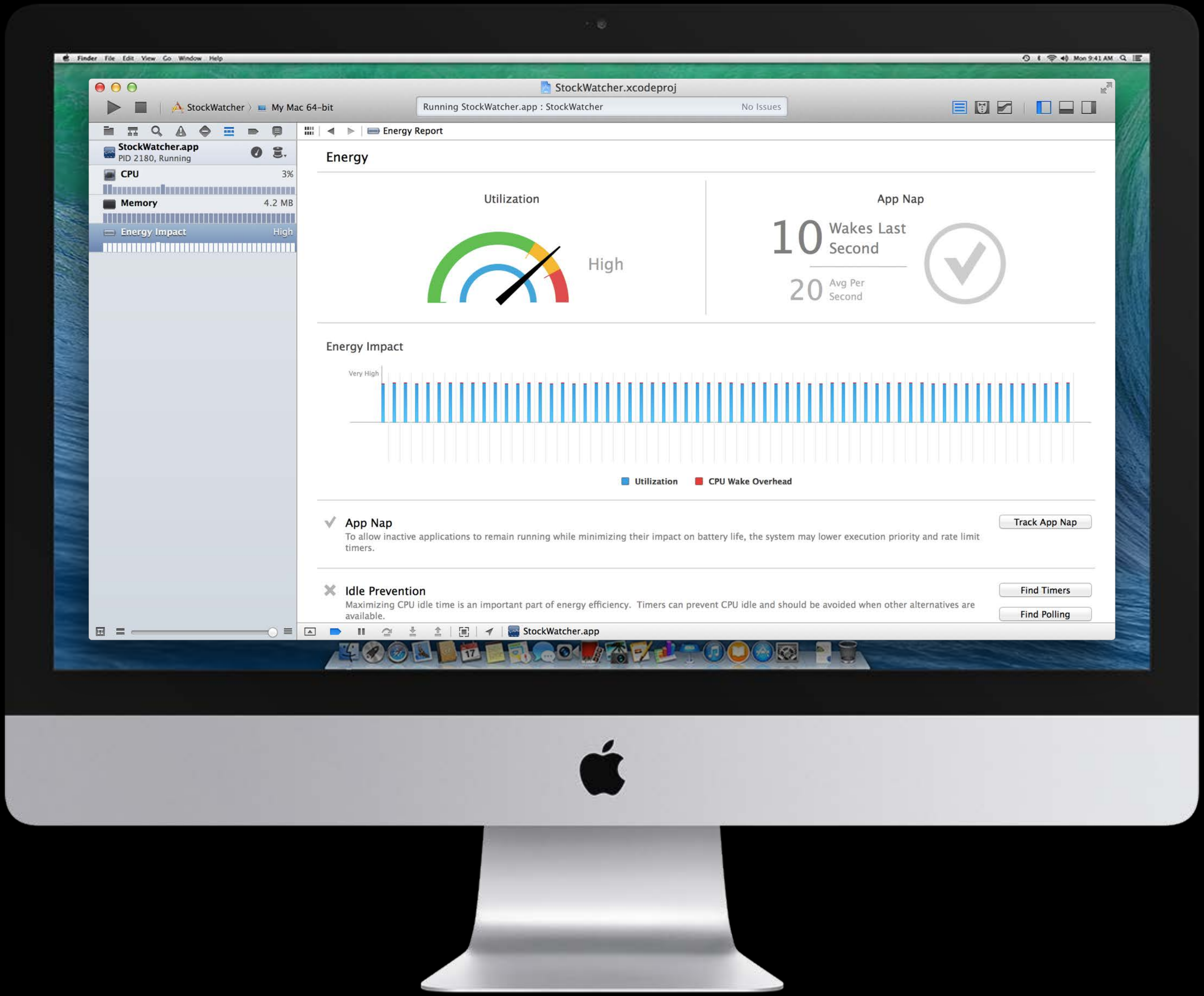
Measuring Impact

Sleep

Final Thoughts

Measuring Impact

Xcode



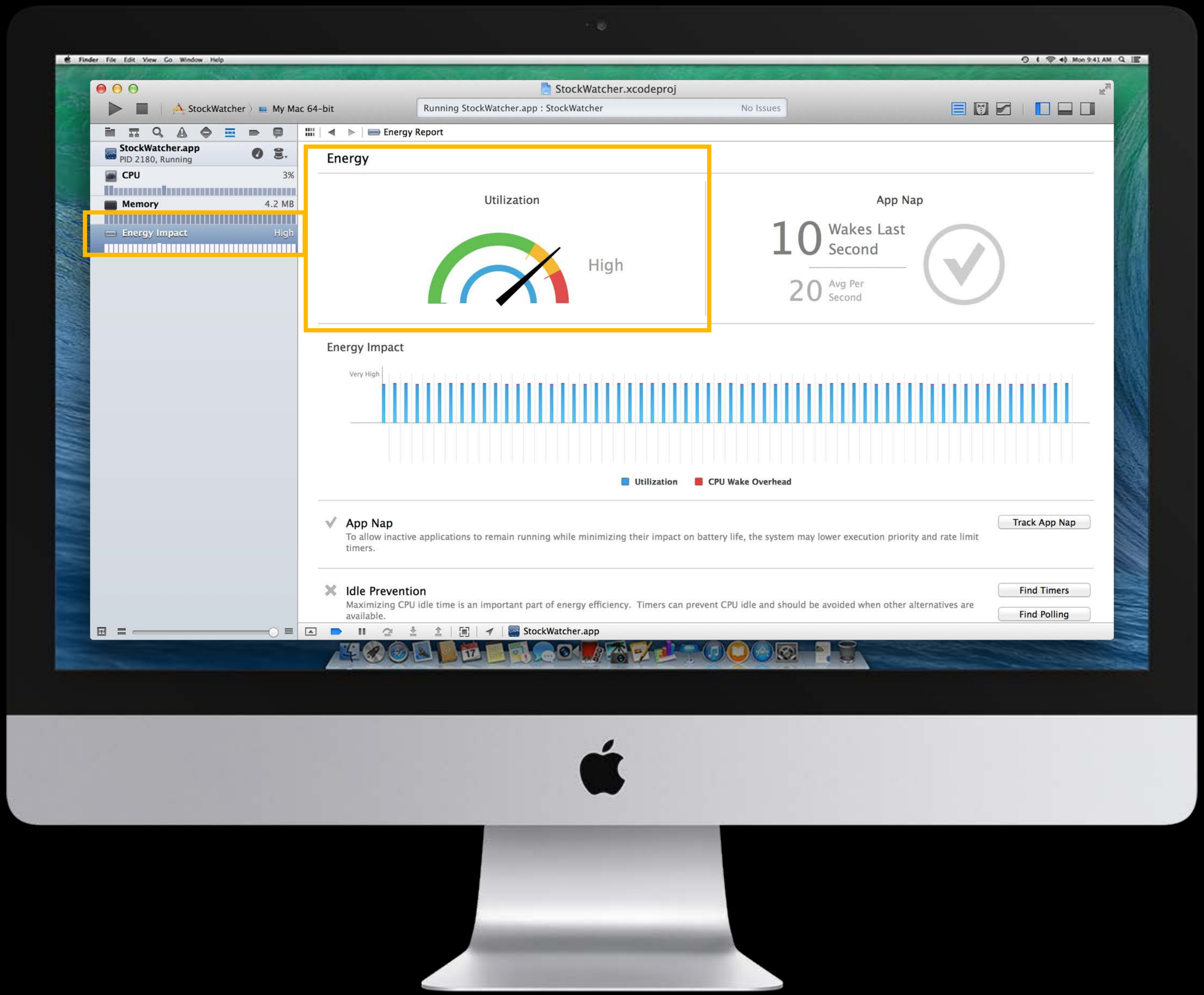
✓ App Nap

To allow inactive applications to remain running while minimizing their impact on battery life, the system may lower execution priority and rate limit timers.

✕ Idle Prevention

Maximizing CPU idle time is an important part of energy efficiency. Timers can prevent CPU idle and should be avoided when other alternatives are available.

Xcode



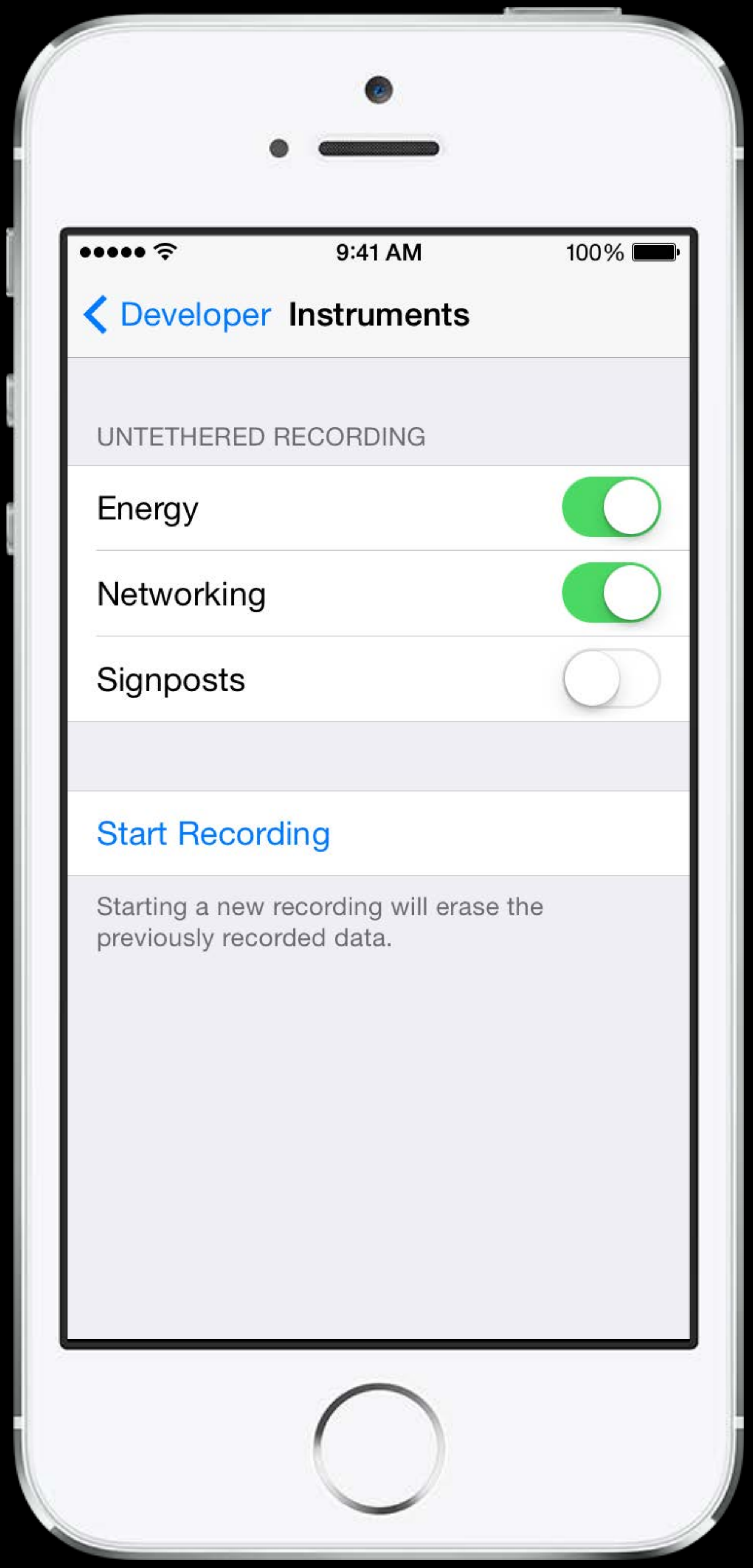
✓ App Nap

To allow inactive applications to remain running while minimizing their impact on battery life, the system may lower execution priority and rate limit timers.

✕ Idle Prevention

Maximizing CPU idle time is an important part of energy efficiency. Timers can prevent CPU idle and should be avoided when other alternatives are available.

Energy Diagnostics



UNTETHERED RECORDING

Energy ●

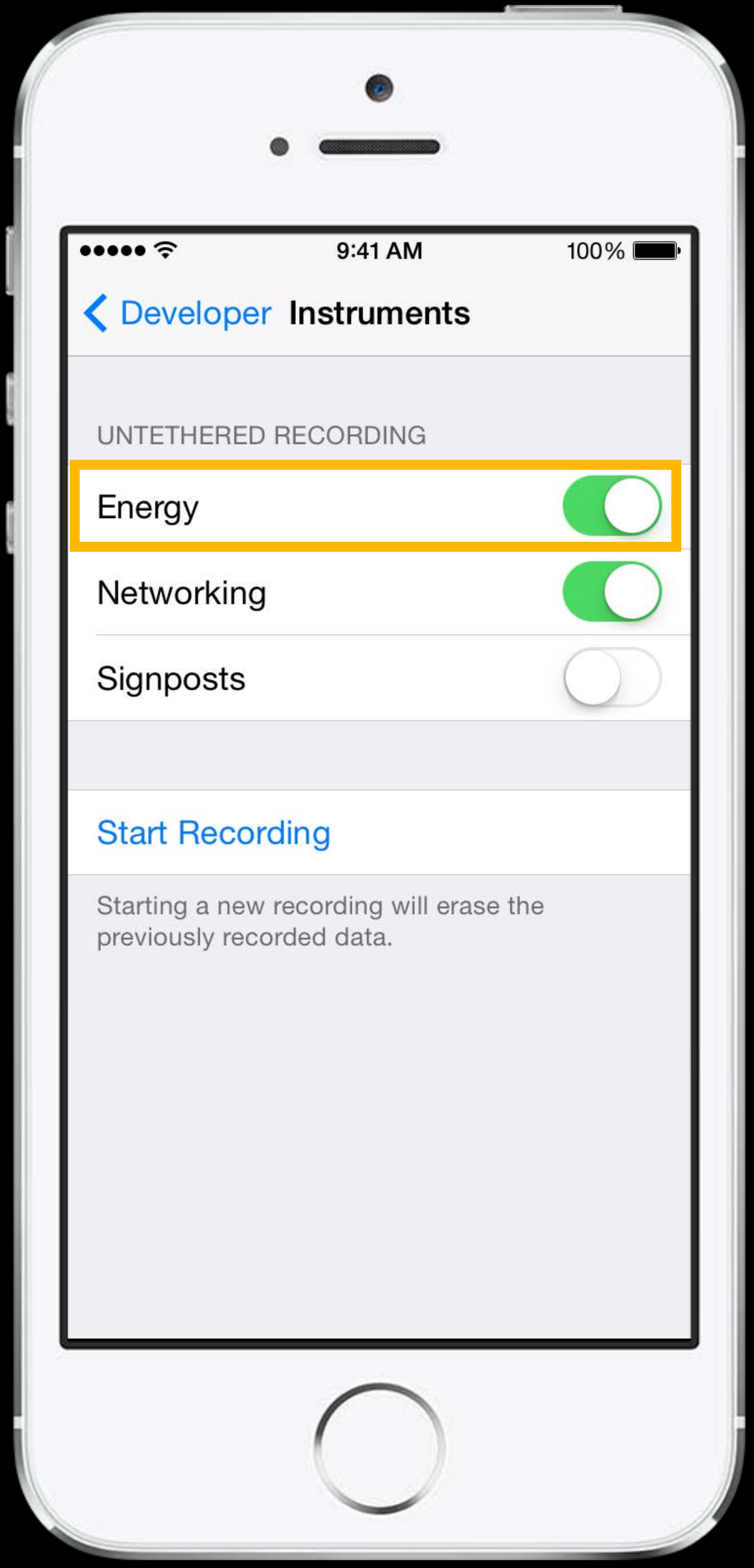
Networking ●

Signposts ○

Start Recording

Starting a new recording will erase the previously recorded data.

Energy Diagnostics



Developer Instruments

UNTETHERED RECORDING

Energy ●

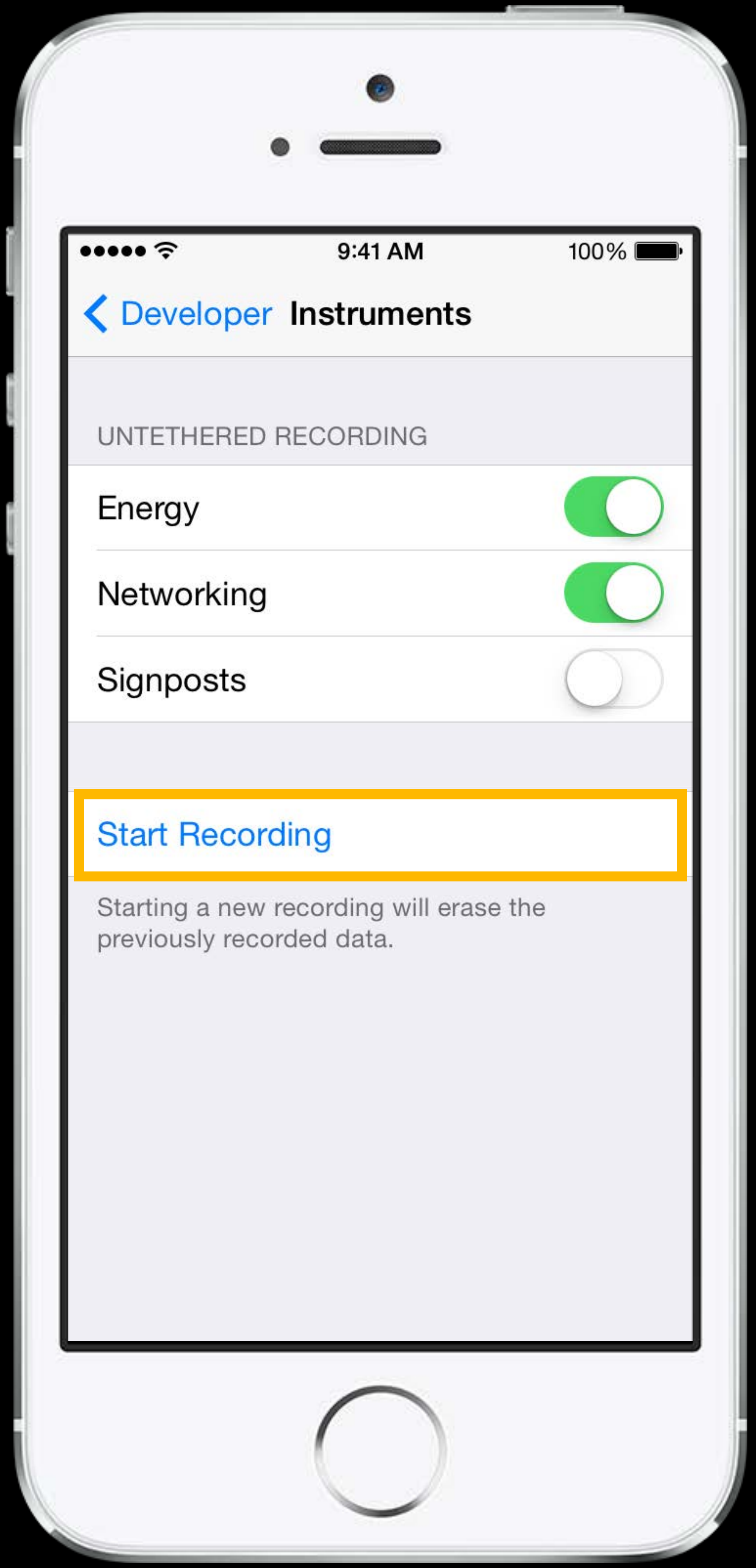
Networking ●

Signposts ○

Start Recording

Starting a new recording will erase the previously recorded data.

Energy Diagnostics



●●●●● 9:41 AM 100%

< Developer **Instruments**

UNTETHERED RECORDING

Energy ●

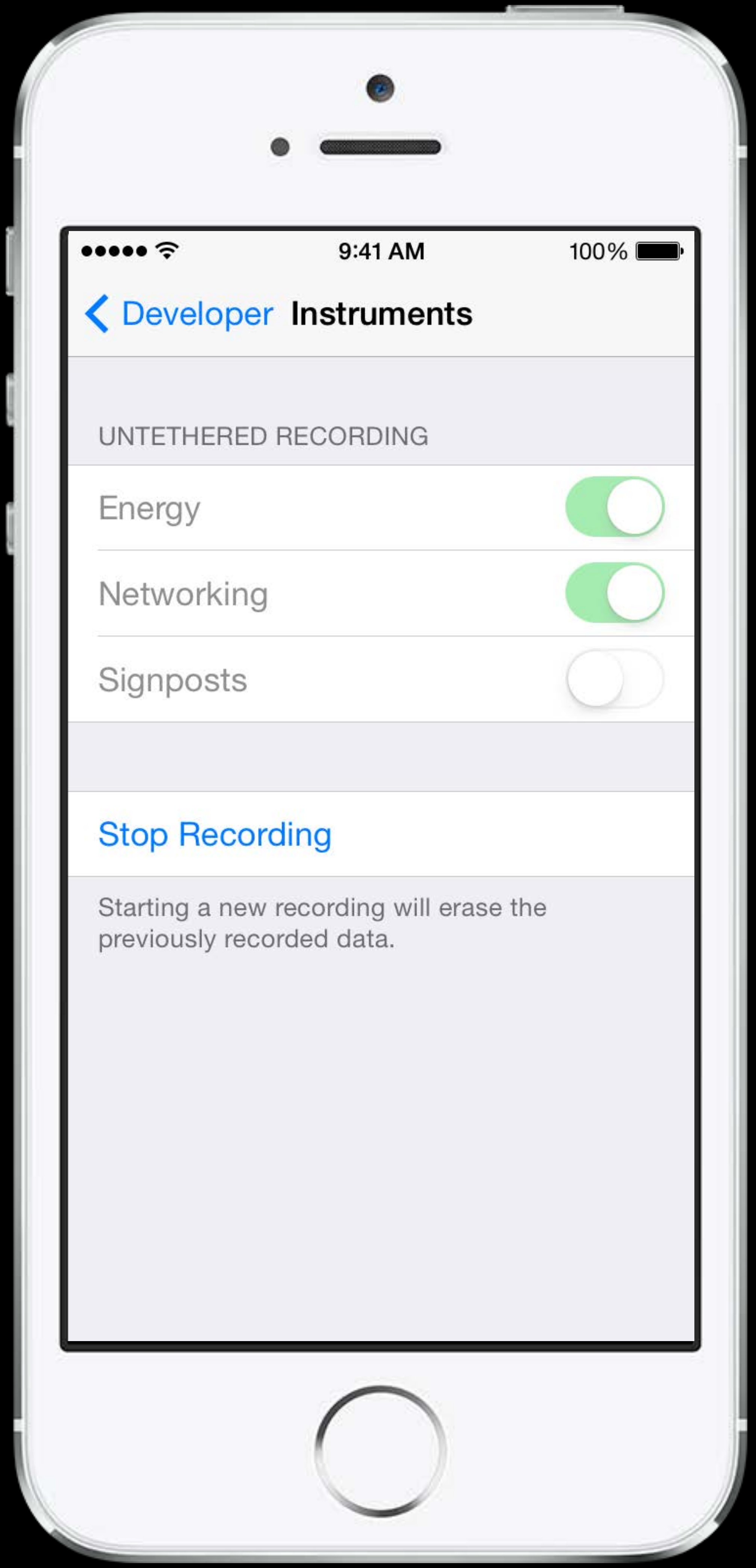
Networking ●

Signposts ○

Start Recording

Starting a new recording will erase the previously recorded data.

Energy Diagnostics



UNTETHERED RECORDING

Energy ●

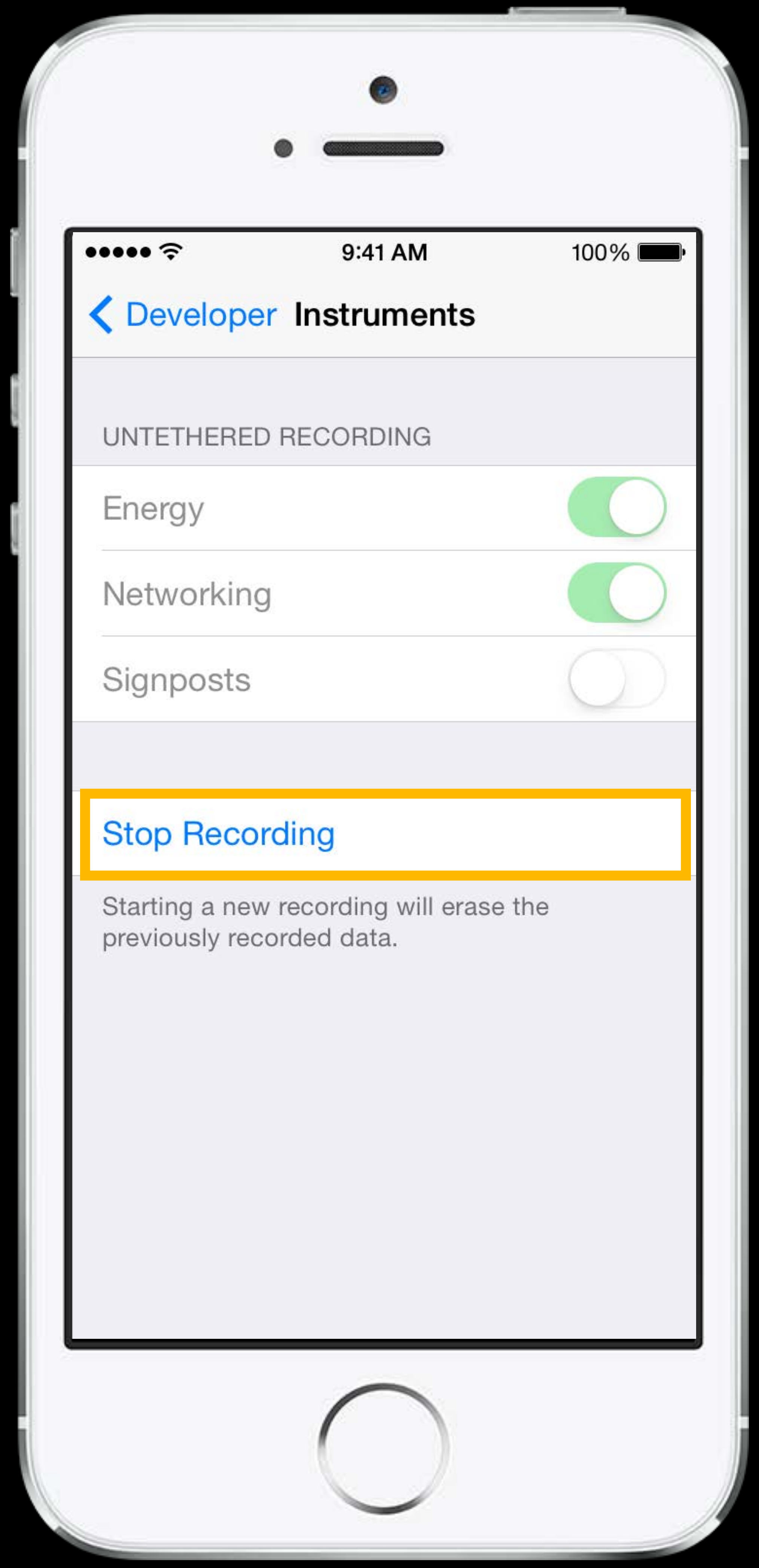
Networking ●

Signposts ○

Stop Recording

Starting a new recording will erase the previously recorded data.

Energy Diagnostics



9:41 AM 100%

Developer Instruments

UNTETHERED RECORDING

Energy ●

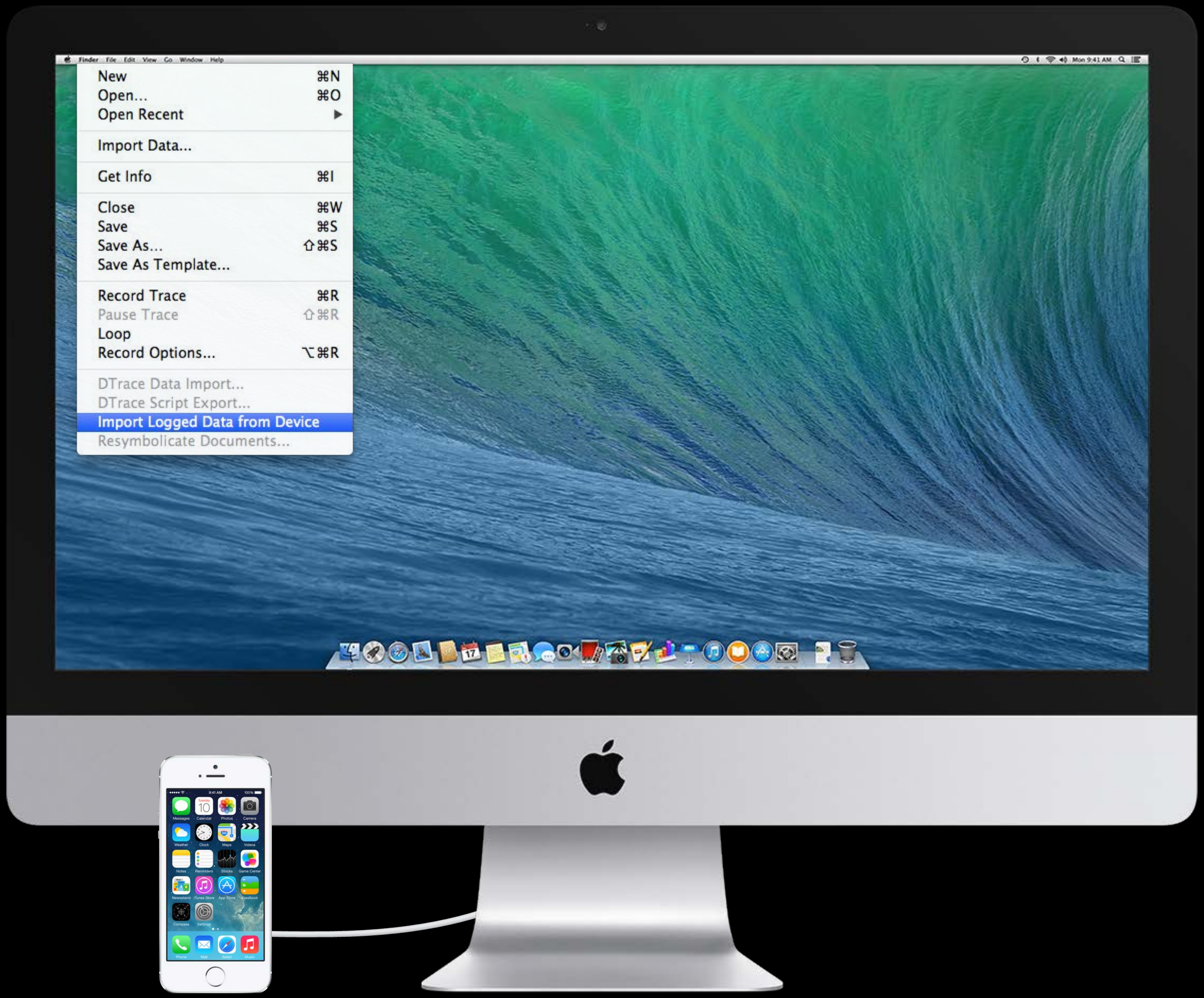
Networking ●

Signposts ○

Stop Recording

Starting a new recording will erase the previously recorded data.

Energy Diagnostics



Demo

Energy diagnostics

Agenda

Quick Recap

Energy Efficient Networking

Measuring Impact

Sleep

Final Thoughts

Sleep

“To sleep is to prepare for the longer journey ahead…”

Sleep

Let the device sleep

Sleep

Let the device sleep

Battery life depends on sleep

Sleep

Let the device sleep

Battery life depends on sleep

iPhone 5S

Sleep

Let the device sleep

Battery life depends on sleep

iPhone 5S

- Web browsing—about 8 to 10 hours

Sleep

Let the device sleep

Battery life depends on sleep

iPhone 5S

- Web browsing—about 8 to 10 hours
- Audio—about 40 hours

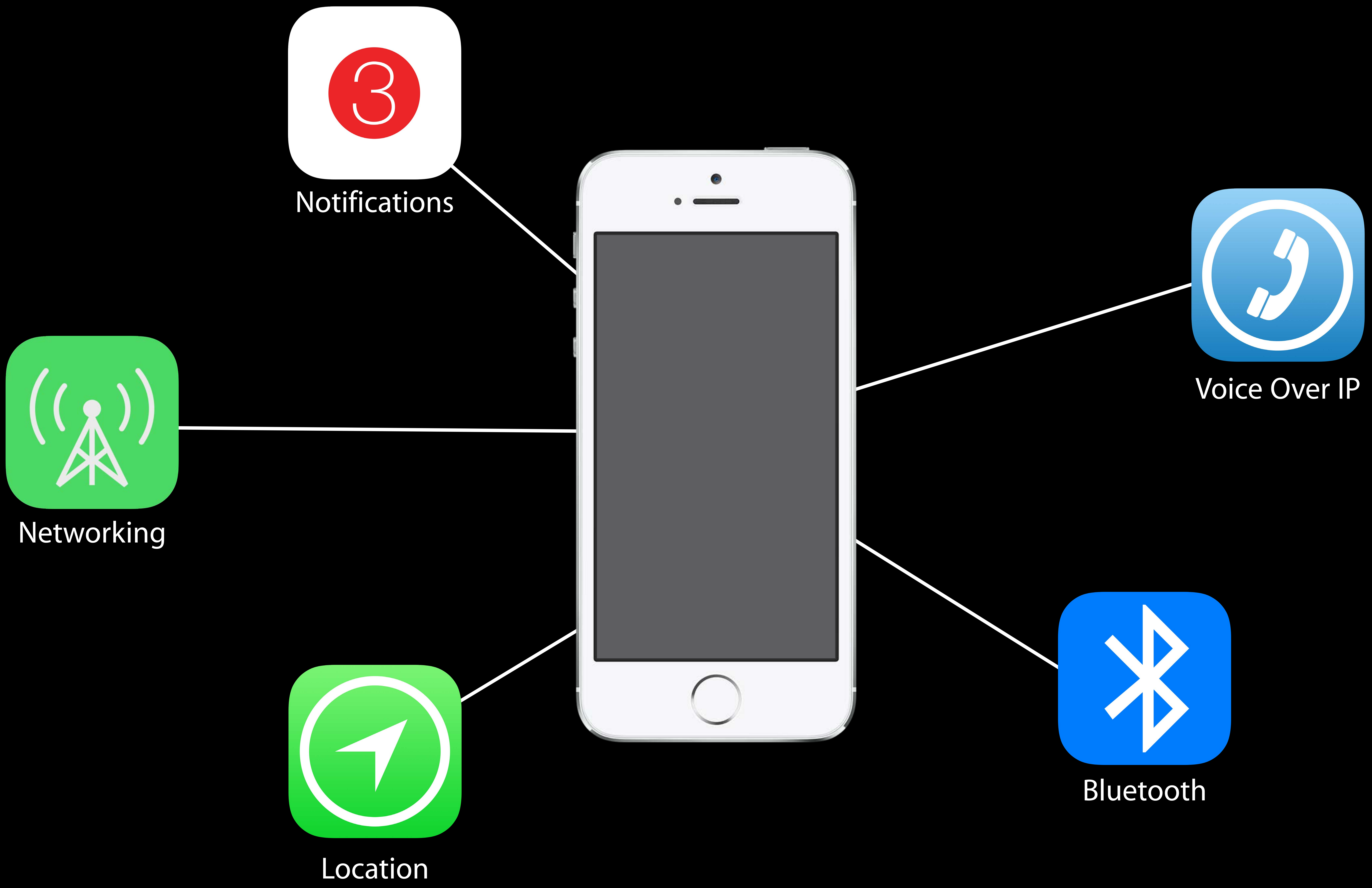
Sleep

Let the device sleep

Battery life depends on sleep

iPhone 5S

- Web browsing—about 8 to 10 hours
- Audio—about 40 hours
- Standby—about 250 hours



Notifications

Networking

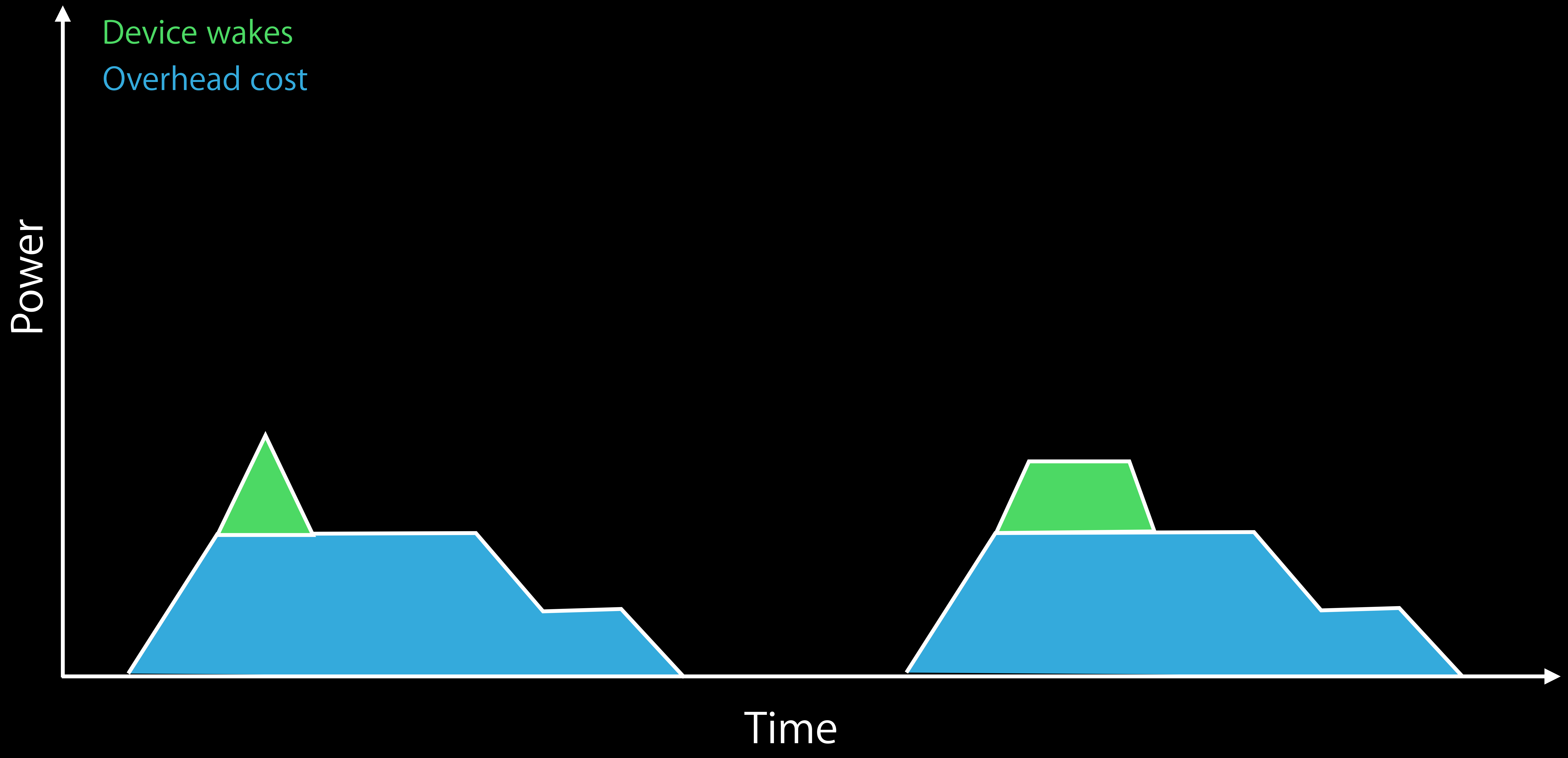
Location

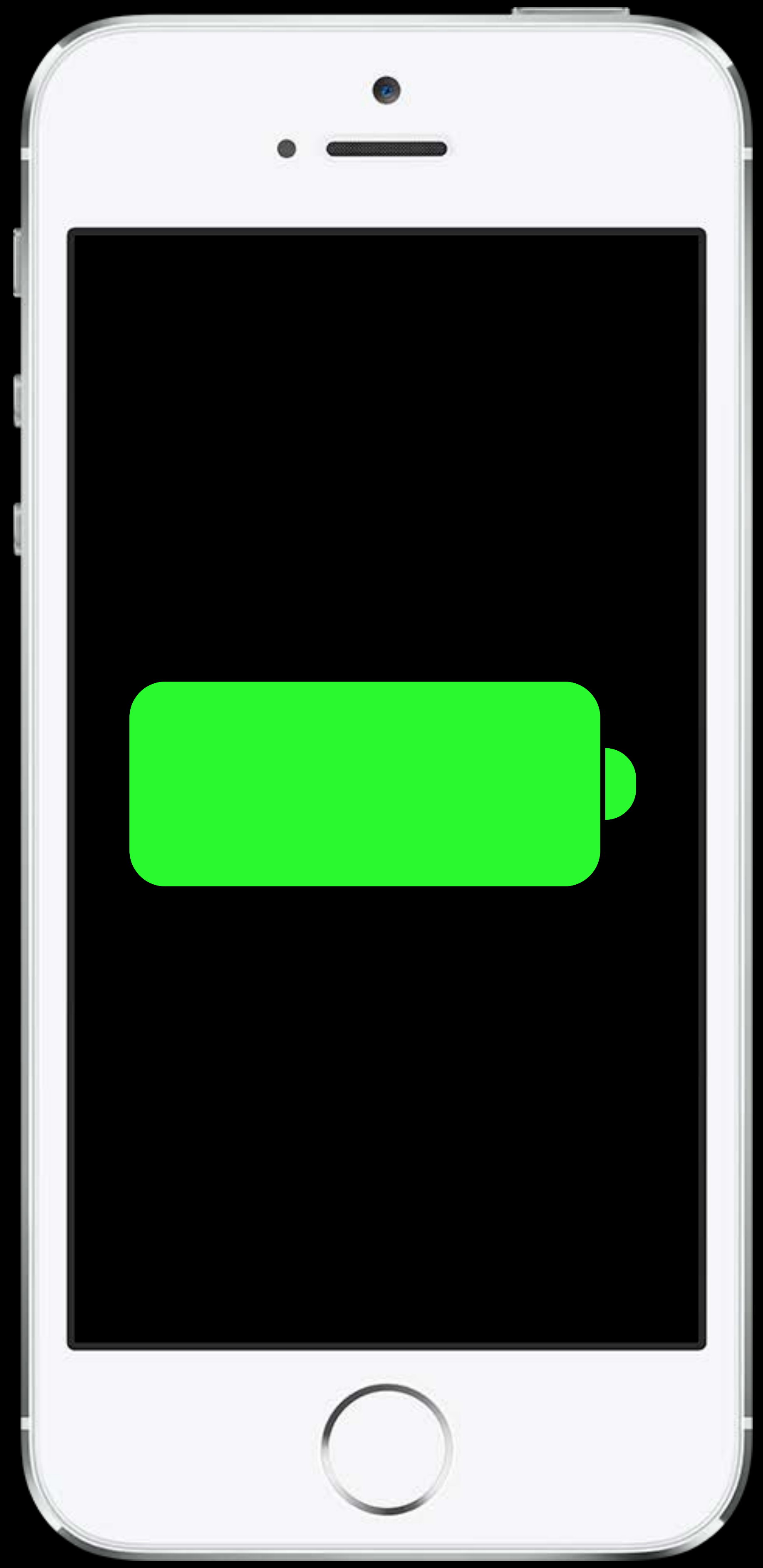
Voice Over IP

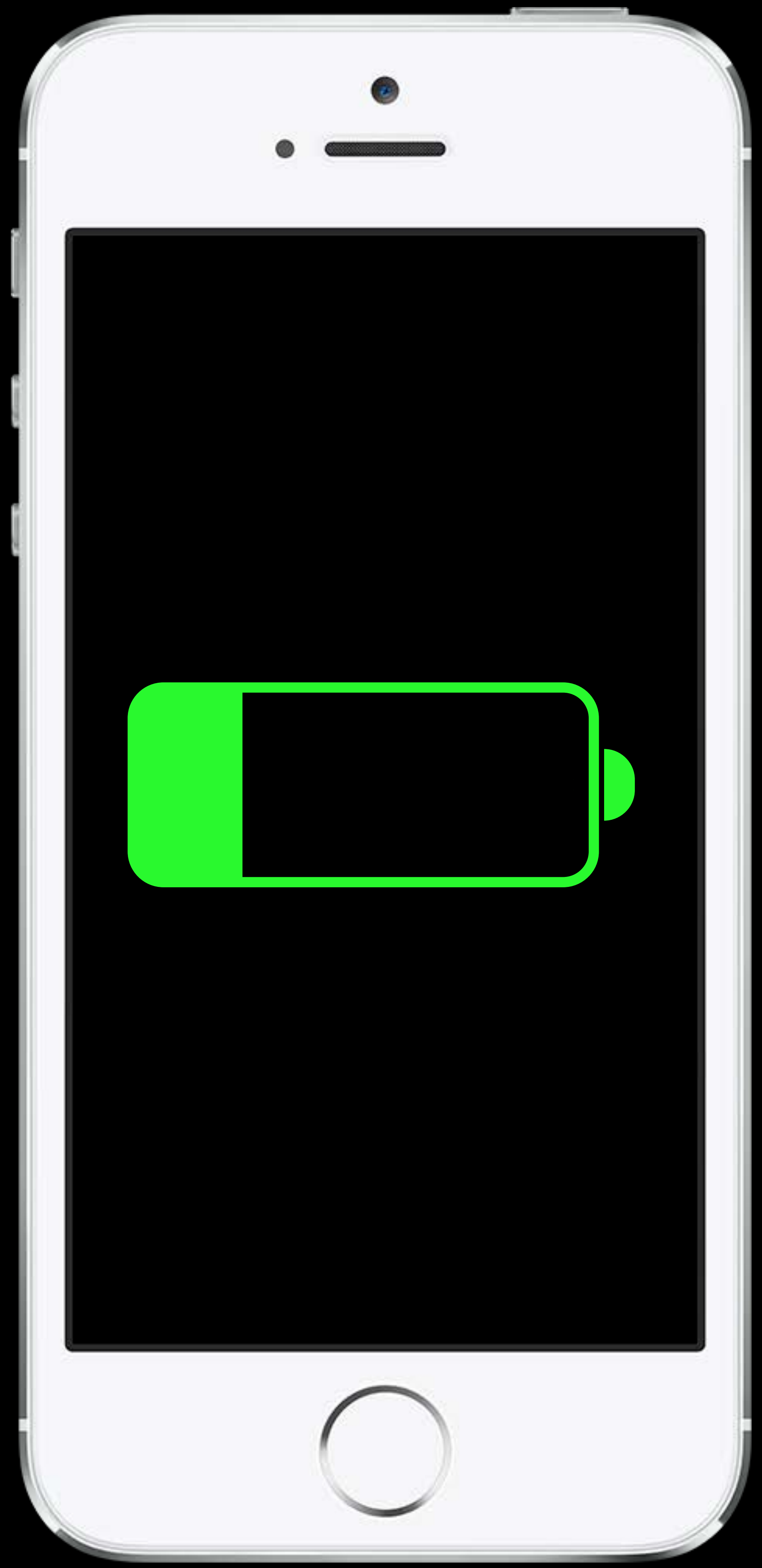
Bluetooth

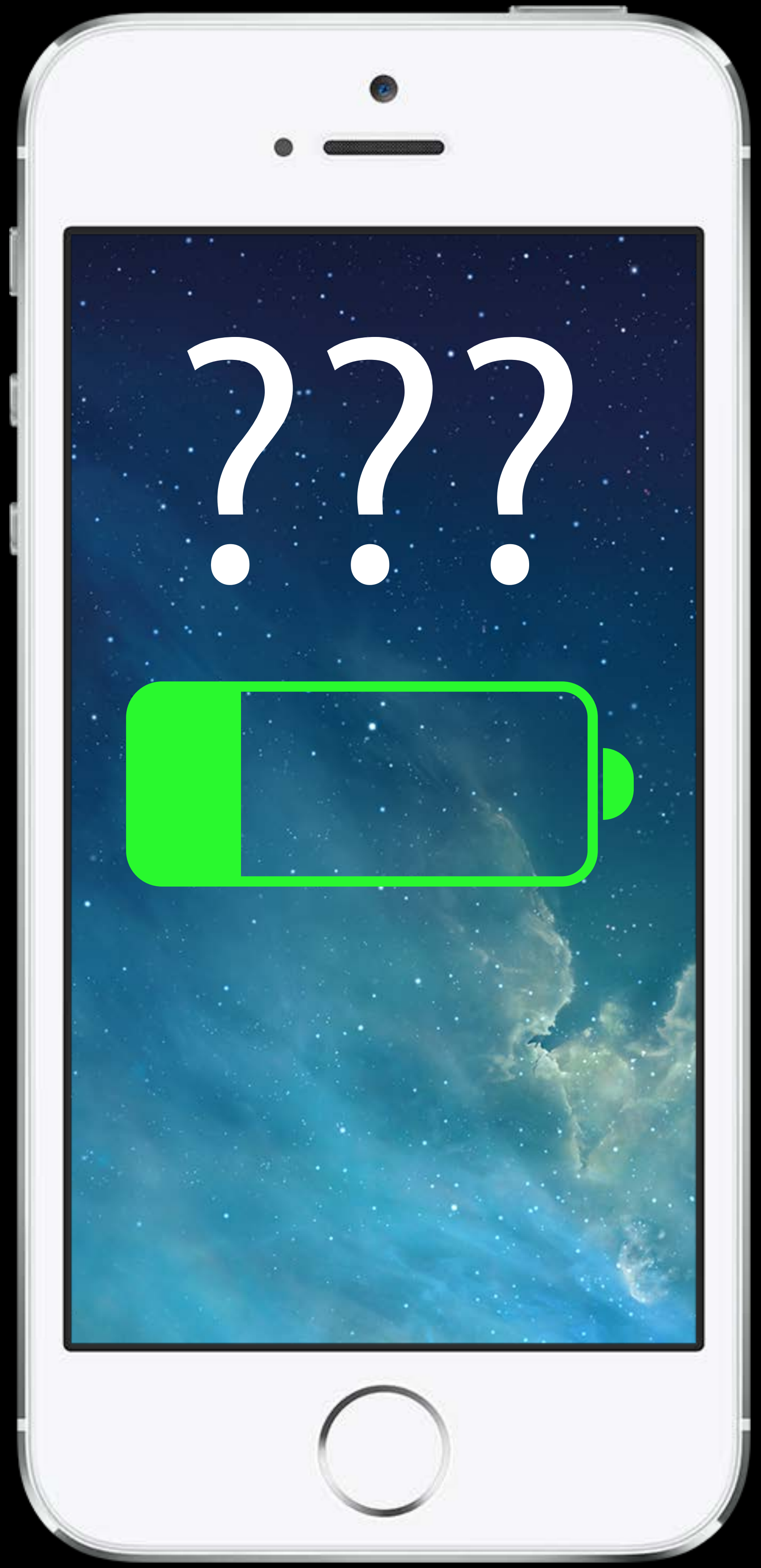
Device wakes

High overhead cost









Sleep/Wakes

Background best practices

Notifications

VoIP

Location

Bluetooth

Notifications

Notifications

Local/Push Notifications

Notifications

Local/Push Notifications

Device will wake for

- Scheduled Local Notifications
- Receive remote Push Notifications

Notifications

Local/Push Notifications

Device will wake for

- Scheduled Local Notifications
- Receive remote Push Notifications

Set Push Notification priority

- 10—Delivered immediately
- 5—Delivered at power conservative time

VoIP

VoIP

VoIP wakes

Persistent connection

- Periodic keep-alive packets
- Device wakes
- Code complexity

VoIP

PushKit framework

Leverages Apple Push Notification Service

- No need to maintain persistent connection

VoIP Push Notifications

- App Runtime to process the pushes
- App Relaunched if terminated
- Include up to 4k payload

VoIP

Application adoption

```
#import <PushKit/PushKit.h>


- (void) voipRegistration {
  PKPushRegistry * voipRegistry = [PKPushRegistry alloc] initWithQueue:
dispatch_get_main_queue()];
  voipRegistry.delegate = self;
  voipRegistry.desiredPushTypes = [NSSet setWithObject:PKPushTypeVoIP]; // register
}
```

VoIP

Application adoption

```
#import <PushKit/PushKit.h>
```

```
- (void) voipRegistration {
  PKPushRegistry * voipRegistry = [PKPushRegistry alloc] initWithQueue:
dispatch_get_main_queue()];
  voipRegistry.delegate = self;
  voipRegistry.desiredPushTypes = [NSSet setWithObject:PKPushTypeVoIP]; // register
}
```

VoIP

Application adoption

```
#import <PushKit/PushKit.h>
```

```
- (void) voipRegistration {
  PKPushRegistry * voipRegistry = [PKPushRegistry alloc] initWithQueue:
dispatch_get_main_queue()];
  voipRegistry.delegate = self;
  voipRegistry.desiredPushTypes = [NSSet setWithObject:PKPushTypeVoIP]; // register
}
```

VoIP

Application adoption

```
– (void)pushRegistry:(PKPushRegistry *)registry didUpdatePushCredentials:
(PKPushCredentials *)credentials forType:(NSString *)type {
  // register voip push token with server
}


– (void)pushRegistry:(PKPushRegistry *)registry didReceiveIncomingPushWithPayload:
(PKPushPayload *)payload forType:(NSString *)type {
  // received push
}
```

VoIP

Application adoption

```
- (void)pushRegistry:(PKPushRegistry *)registry didUpdatePushCredentials:
(PKPushCredentials *)credentials forType:(NSString *)type {

  // register voip push token with server

}
```

```
- (void)pushRegistry:(PKPushRegistry *)registry didReceiveIncomingPushWithPayload:
(PKPushPayload *)payload forType:(NSString *)type {

  // received push

}
```

VoIP

Application adoption

```
- (void)pushRegistry:(PKPushRegistry *)registry didUpdatePushCredentials:
(PKPushCredentials *)credentials forType:(NSString *)type {
  // register voip push token with server
}
```

```
- (void)pushRegistry:(PKPushRegistry *)registry didReceiveIncomingPushWithPayload:
(PKPushPayload *)payload forType:(NSString *)type {
  // received push
}
```

VoIP

Server side changes

Request VoIP Push Certificate

Send Push with VoIP Certificate

- Receiving device requires iOS 8

Location

Location

Continuous location updates

Location

Continuous location updates

[locationManager startUpdatingLocation]

Prevent device sleep

Accuracy makes a difference

Energy Efficient Location

Deferred location updates

Energy Efficient Location

Deferred location updates

[locationManager allowDeferredLocationUpdatesUntilTraveled: timeout:]

Available for GPS level accuracy

Data buffered on location hardware

Example

- Run tracking app

Energy Efficient Location

Significant location change

Energy Efficient Location

Significant location change

[locationManager startMonitoringSignificantLocationChanges]

Location data with filters

- Distance—Over 500 meters
- Time—Over 5 minutes

Example

- Weather app

Energy Efficient Location

Region monitoring

Energy Efficient Location

Region monitoring

*[locationManager startMonitoringForRegion:(CLRegion *)]*

Application is notified on

- Entrance into region
- Exit out of region

Example

- Museum app

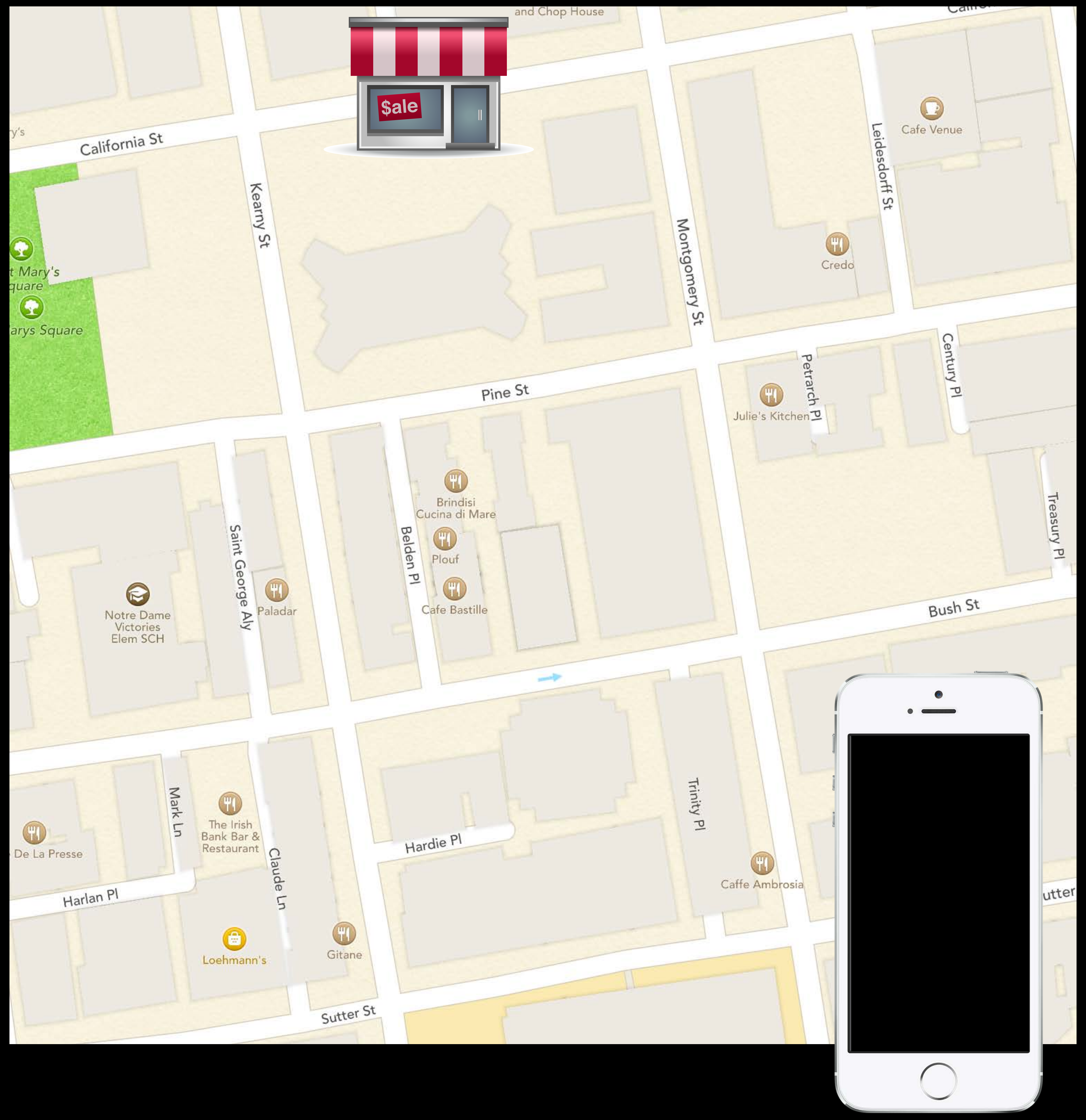
Stop Location Updates

Location

Run tracking app

Requirements

- Keep track of the route
- Update coupons when close to stores

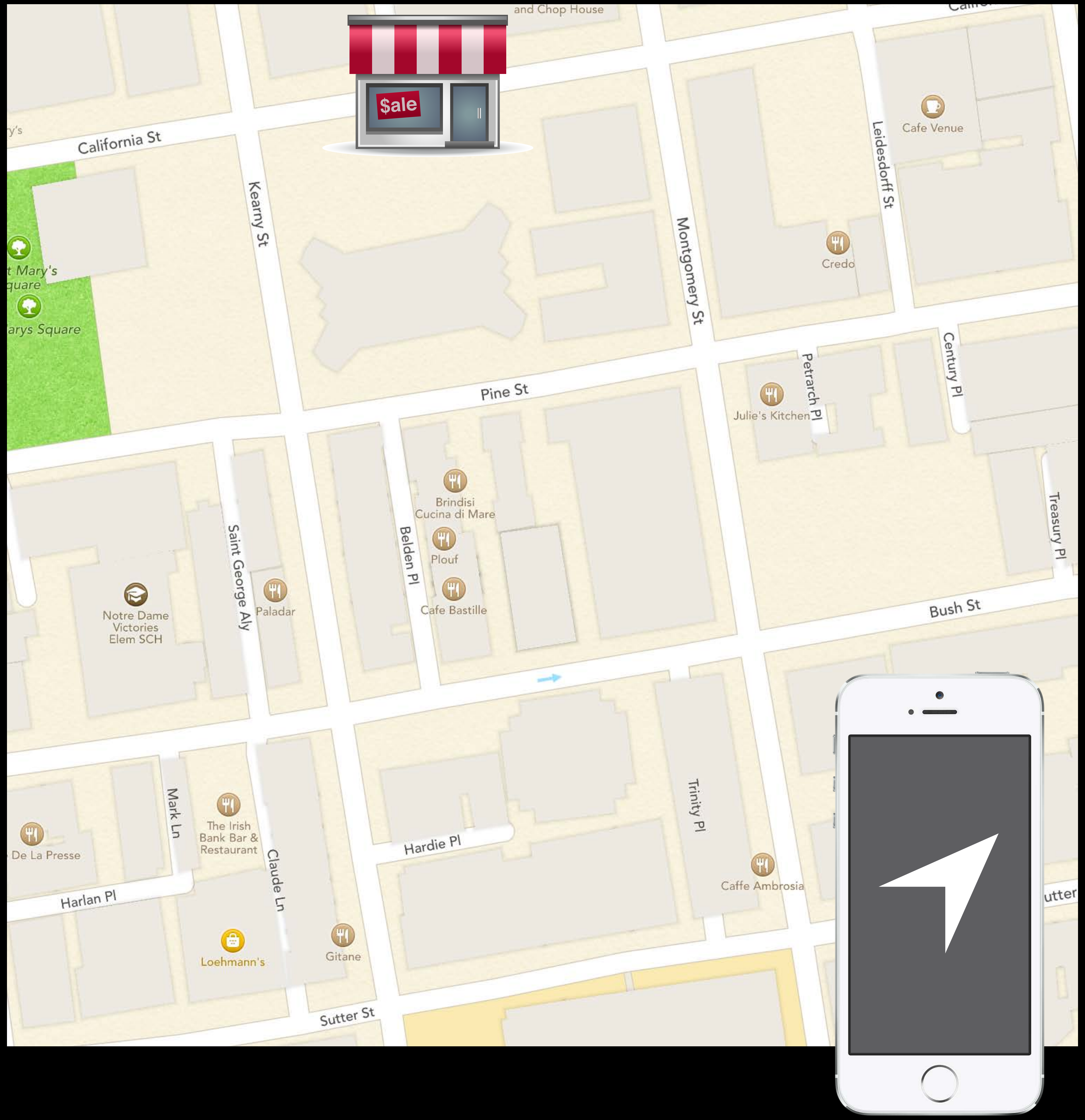


Location

Run tracking app

Current solution

- Continuous location updates

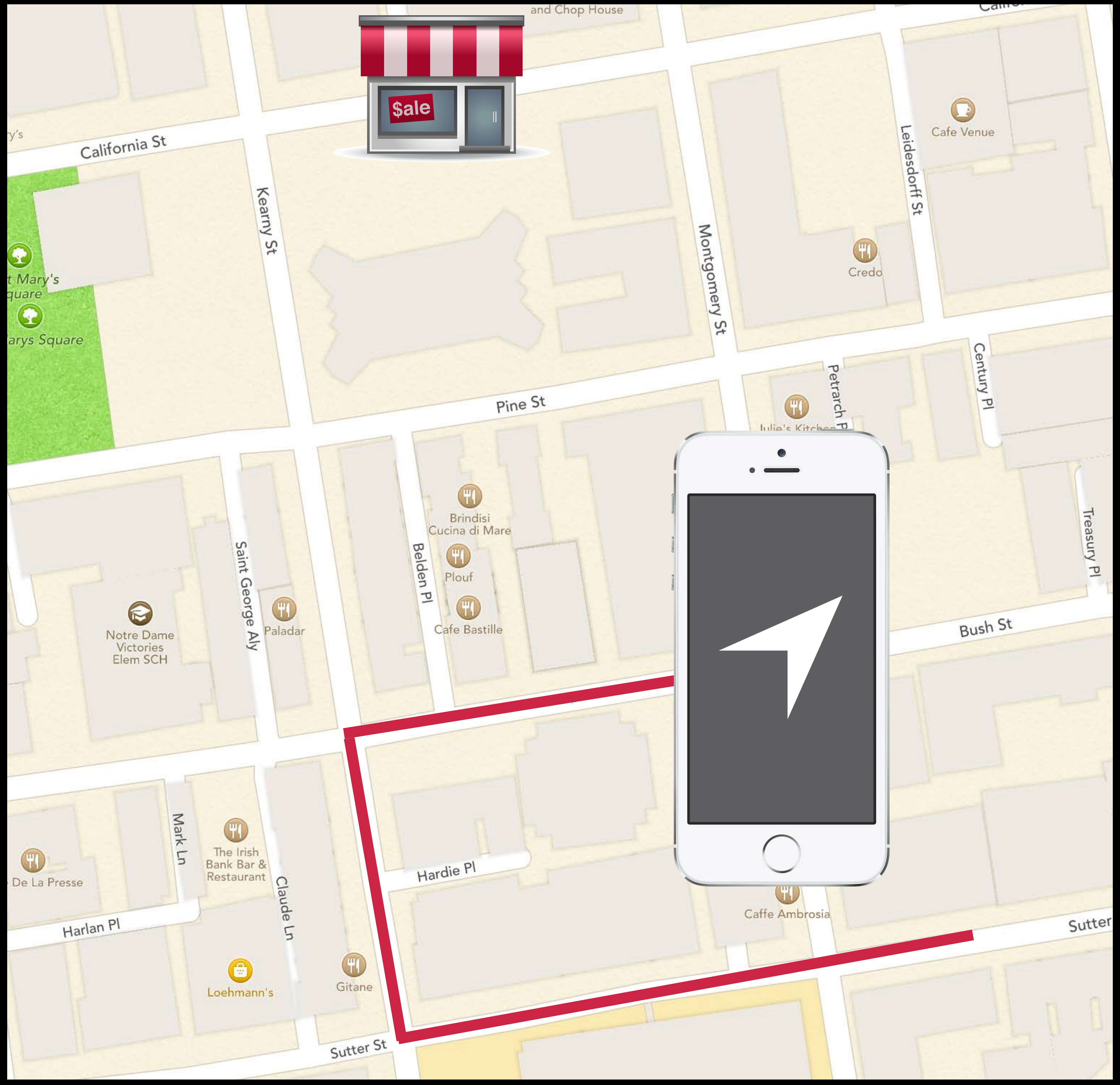


Location

Run tracking app

Current solution

- Continuous location updates

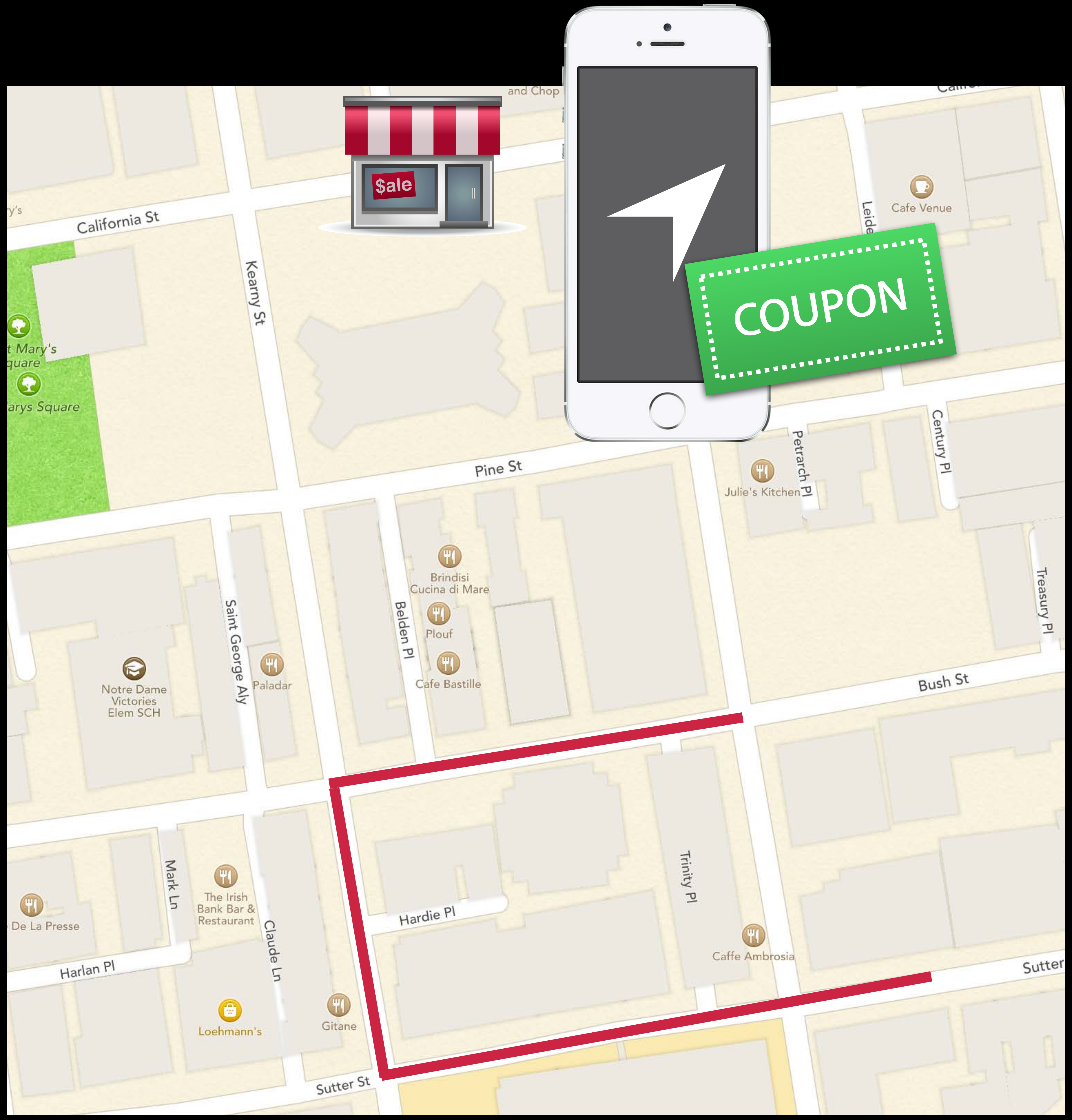


Location

Run tracking app

Current solution

- Continuous location updates

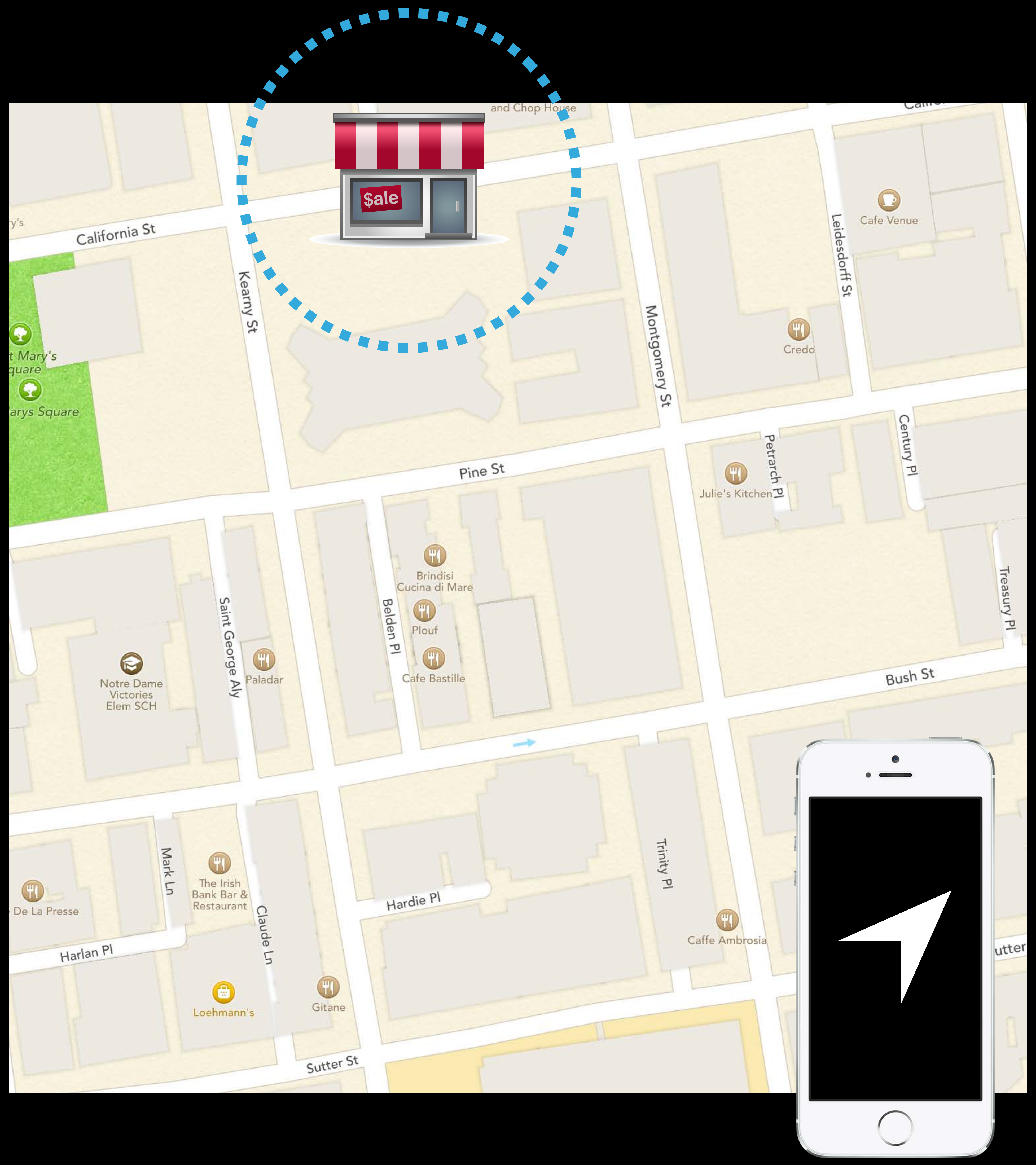


Location

Run tracking app

Energy efficient solution

- Deferred location updates
- Region monitoring

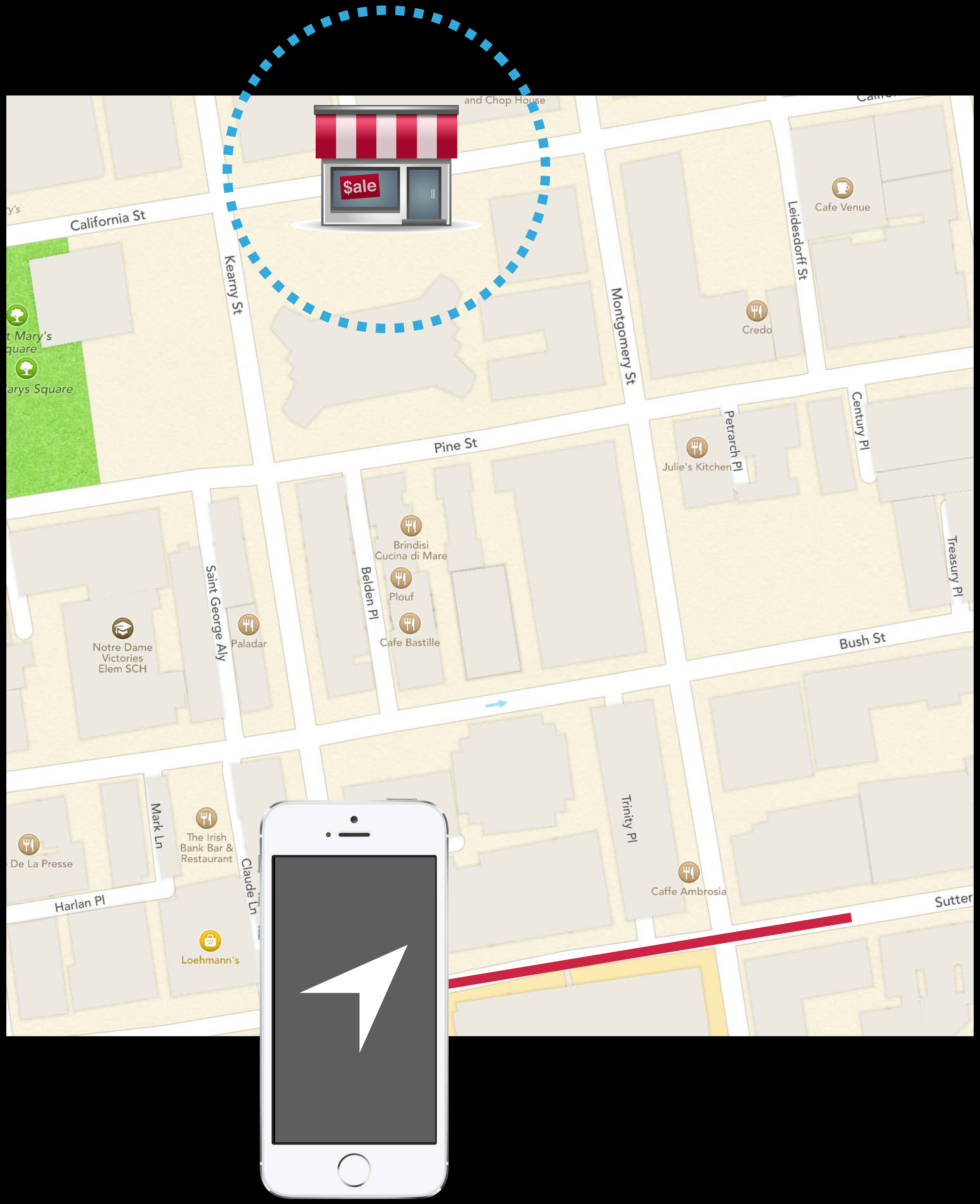


Location

Run tracking app

Energy efficient solution

- Deferred location updates
- Region monitoring

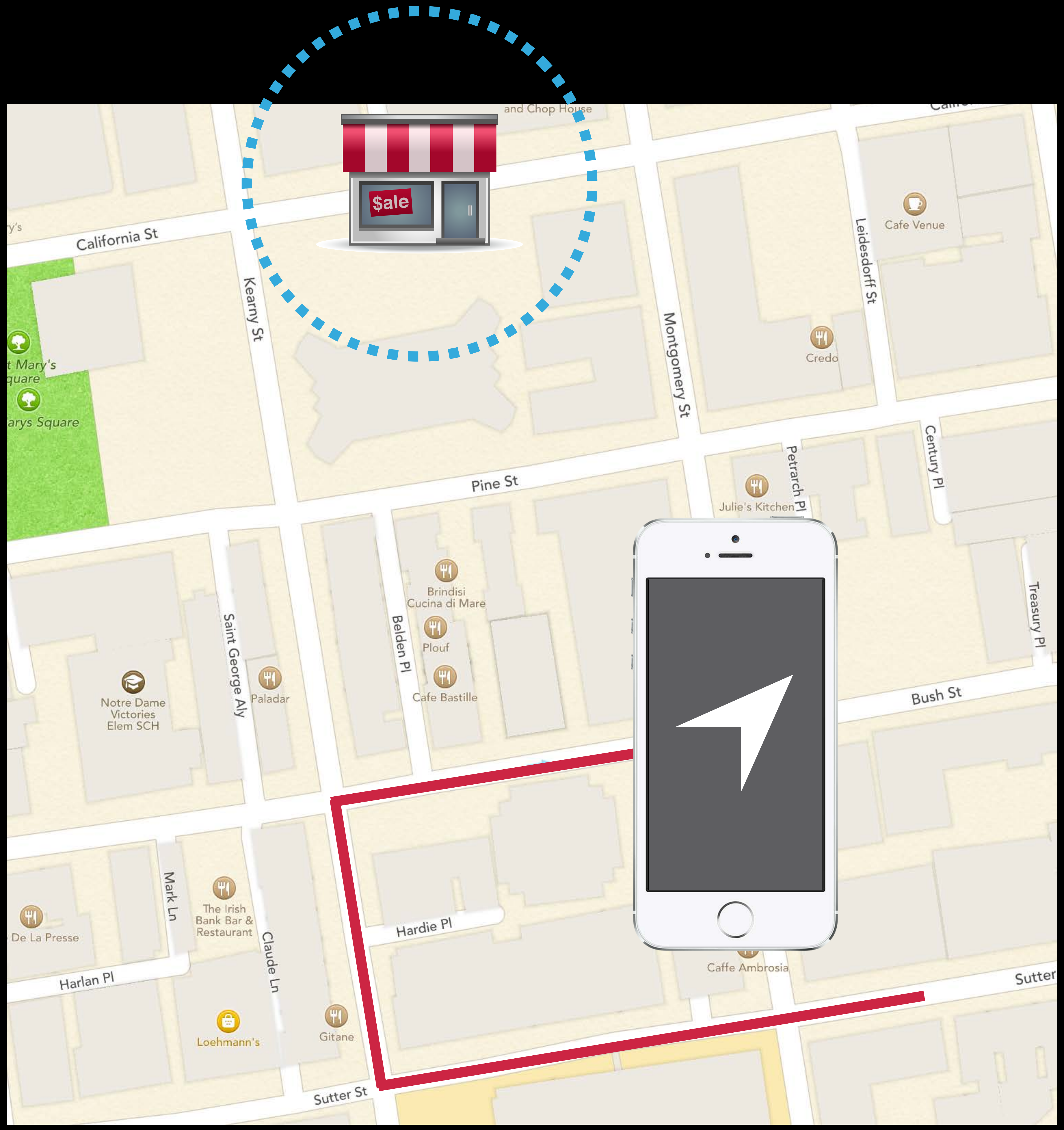


Location

Run tracking app

Energy efficient solution

- Deferred location updates
- Region monitoring

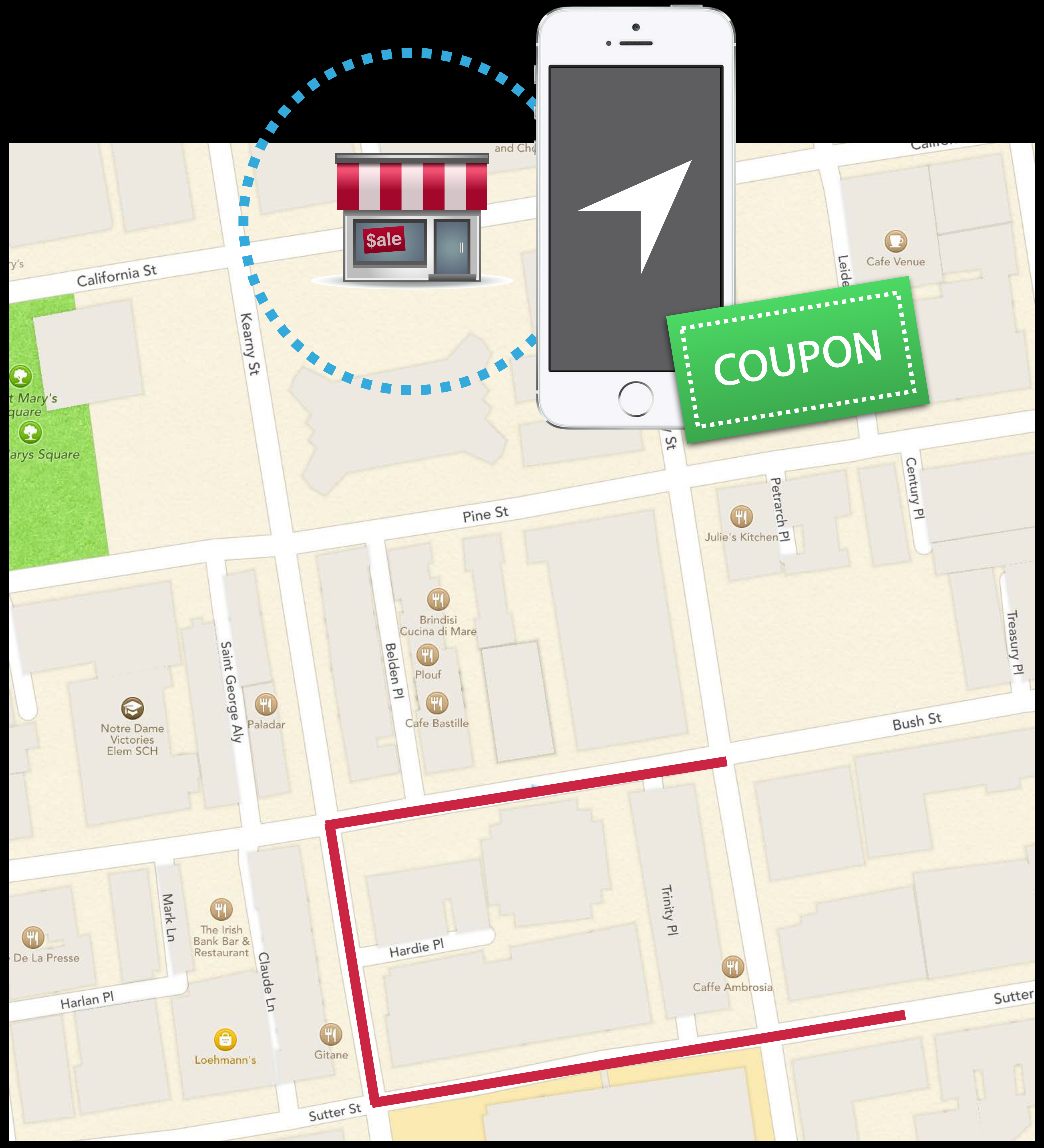


Location

Run tracking app

Energy efficient solution

- Deferred location updates
- Region monitoring



Location

Energy efficient APIs

Deferred location updates

Significant location change

Region monitoring

iBeacons

AutoPause

Location

Energy efficient APIs

NEW

Deferred location updates

Significant location change

Region monitoring

iBeacons

AutoPause

Significant locations visited

Bluetooth

Bluetooth

Interacting with accessories

Bluetooth

Interacting with accessories

Data transfer

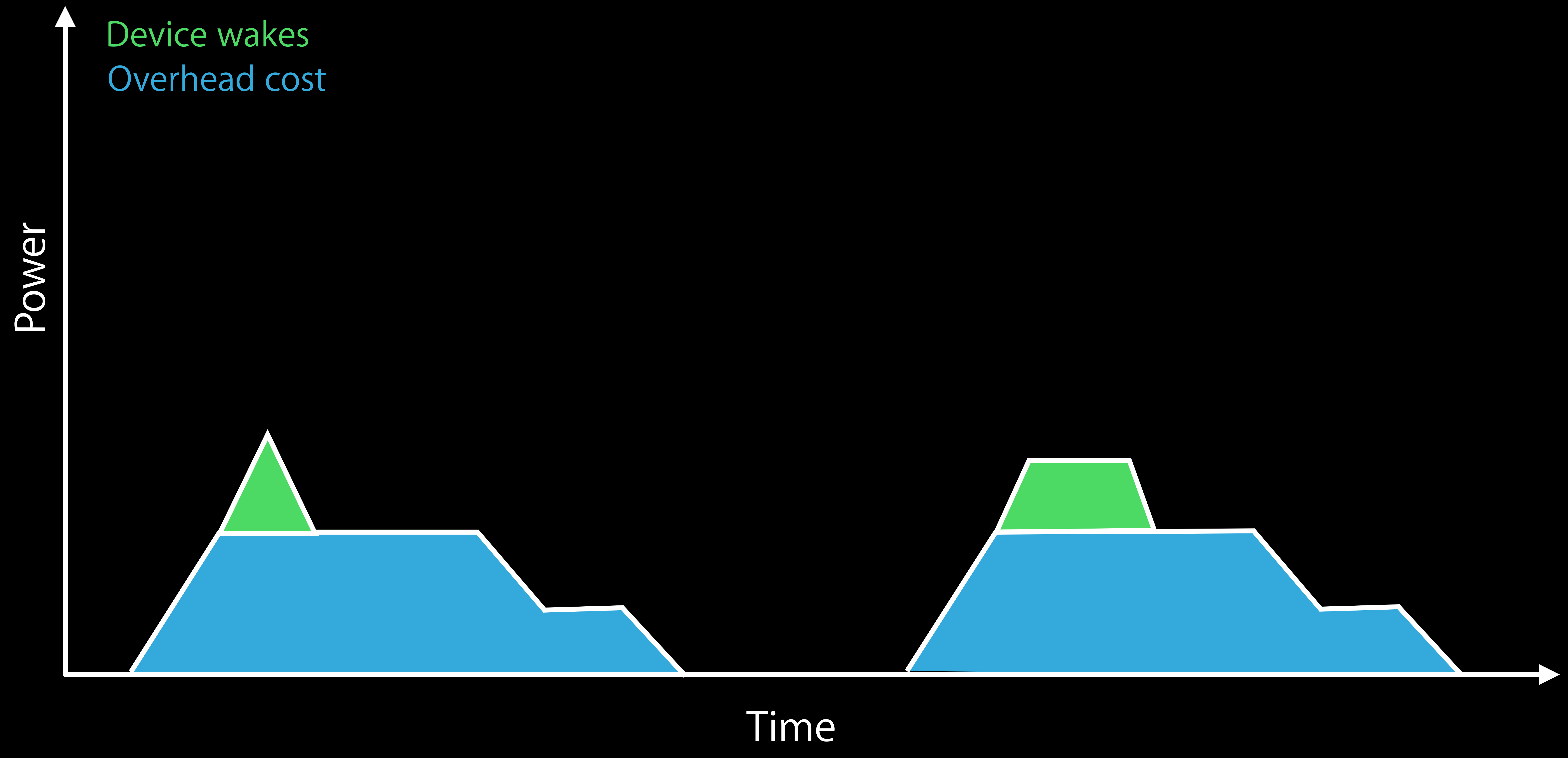
- Device wake for application

Bluetooth

Interacting with accessories

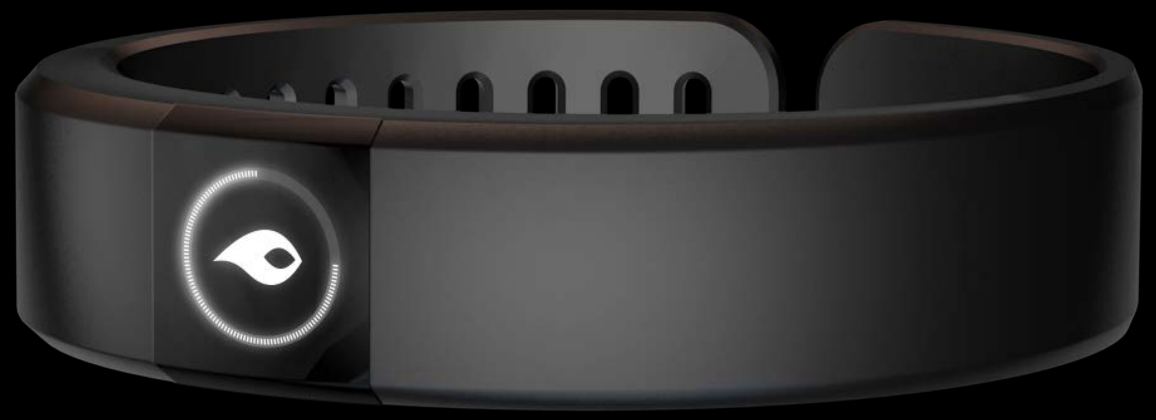
Data transfer

- Device wake for application



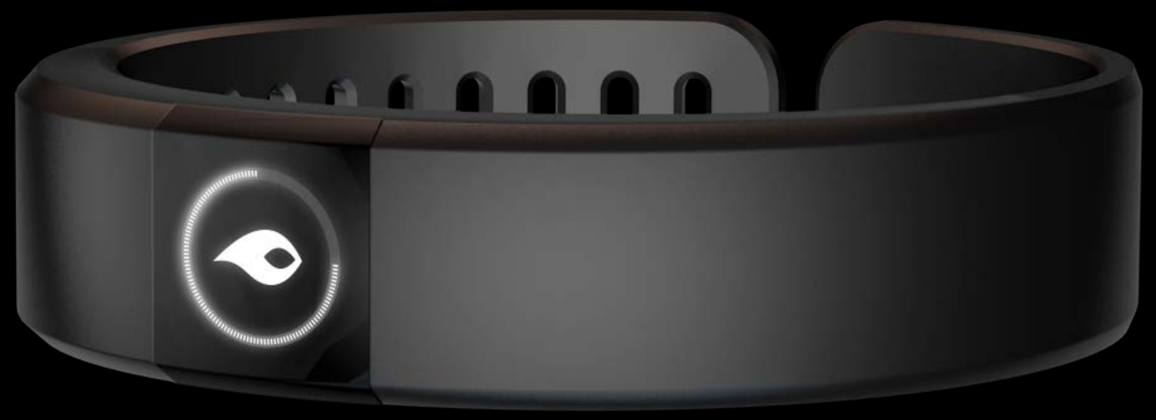
Bluetooth

Run tracking app



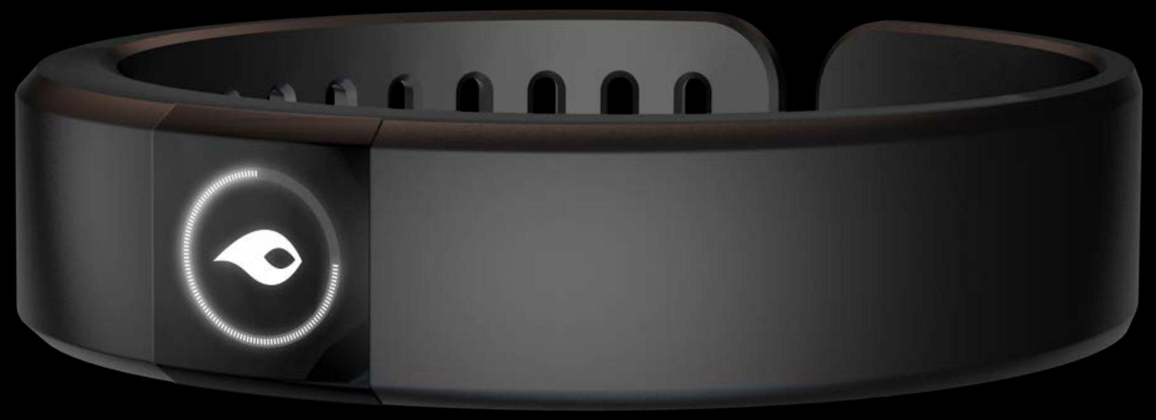
Bluetooth

Streaming



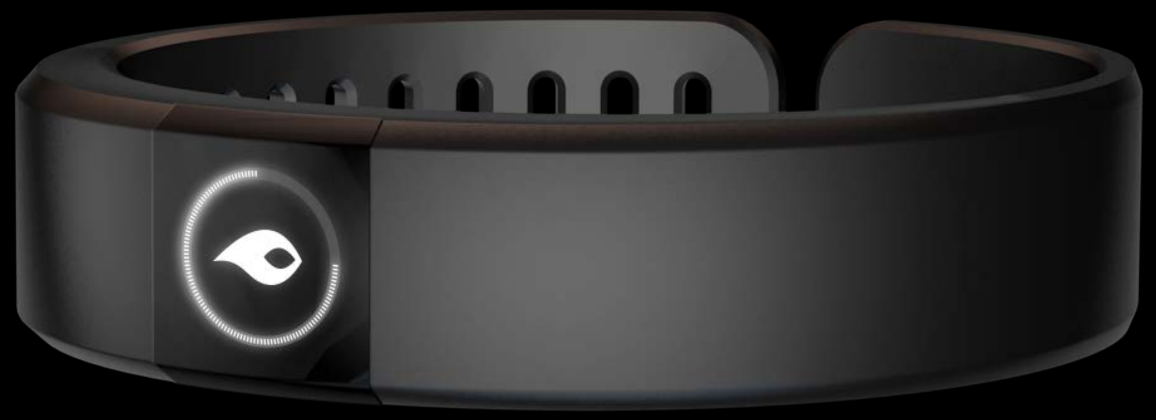
Bluetooth

Streaming



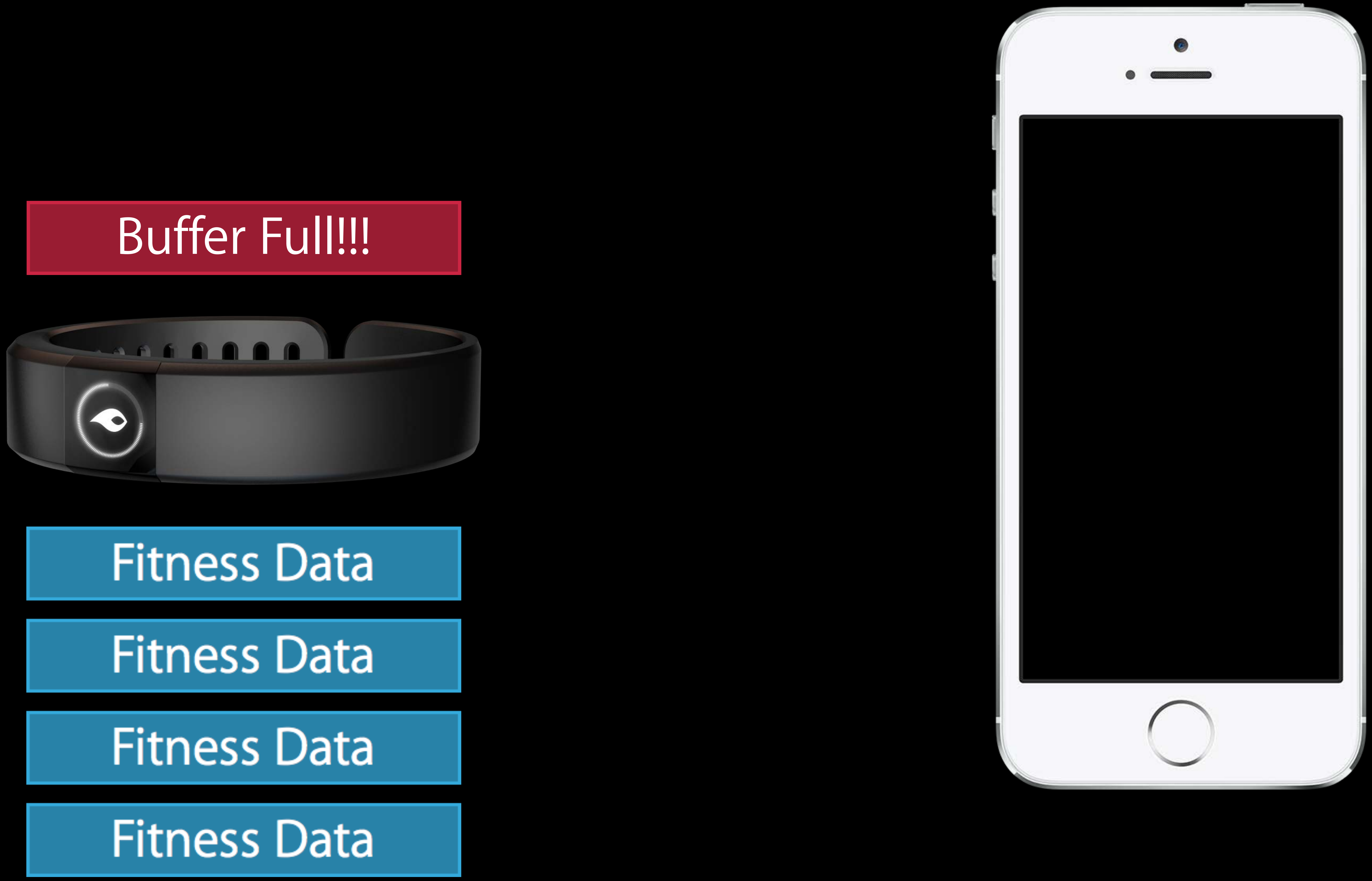
Bluetooth

Buffering



Bluetooth

Buffering



Buffer Full!!!

Fitness Data

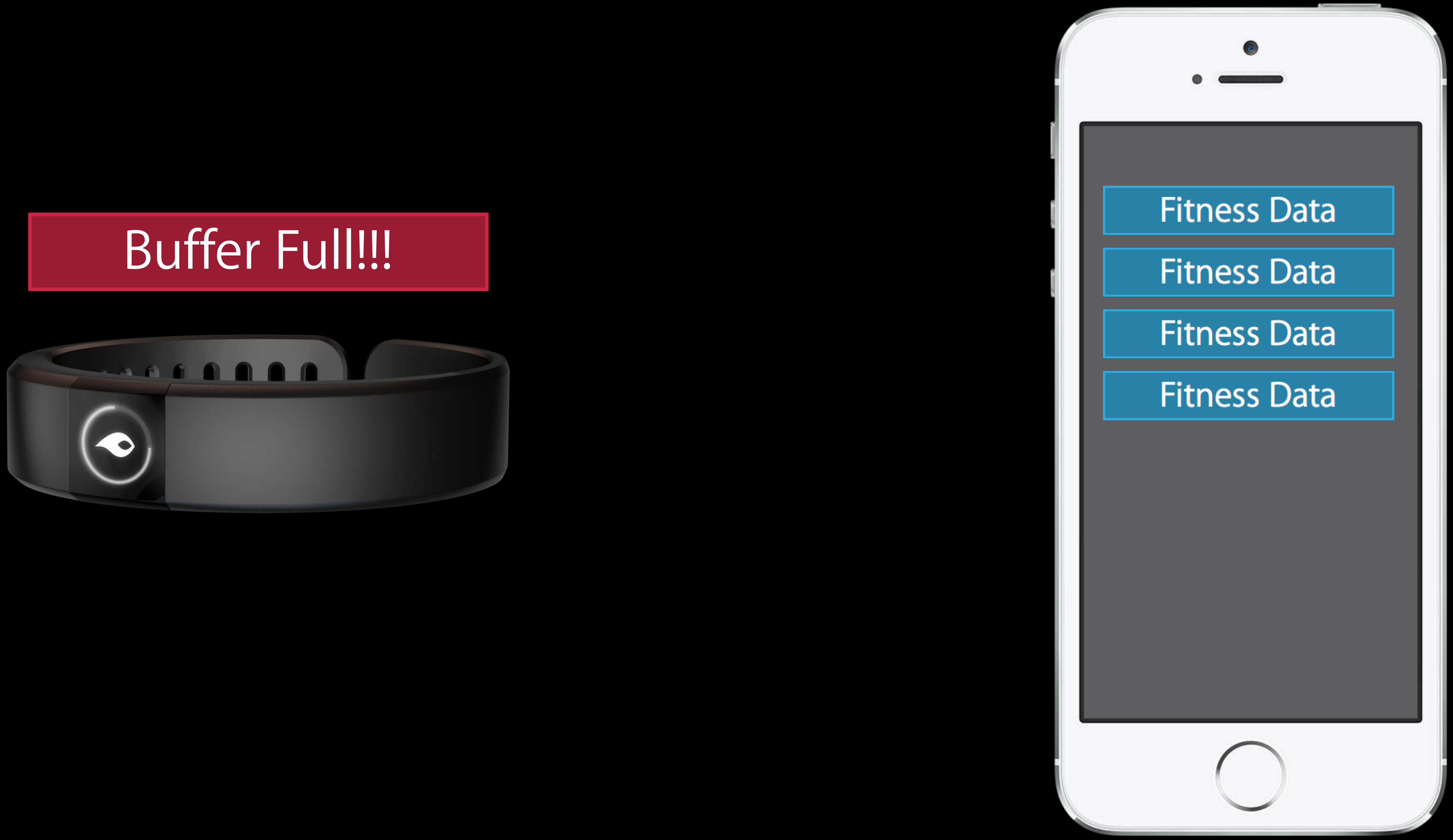
Fitness Data

Fitness Data

Fitness Data

Bluetooth

Buffering



Buffer Full!!!

Fitness Data

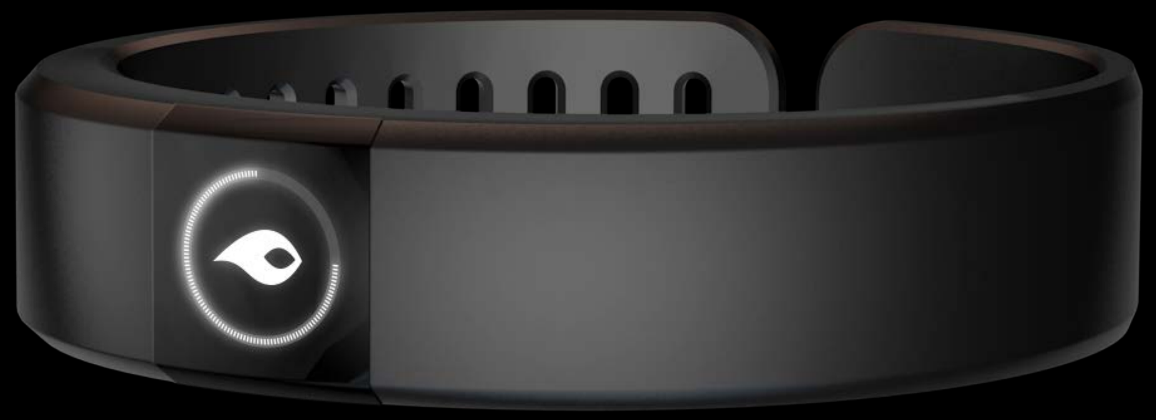
Fitness Data

Fitness Data

Fitness Data

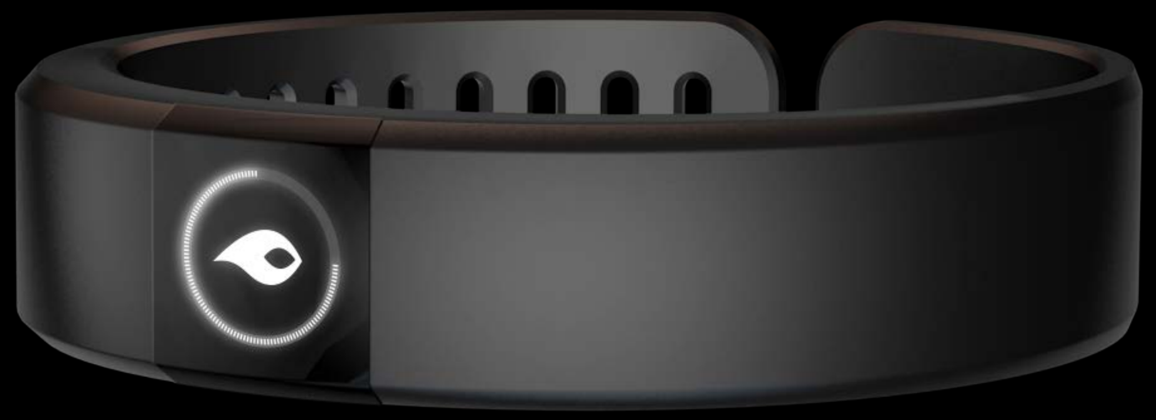
Bluetooth

Buffering



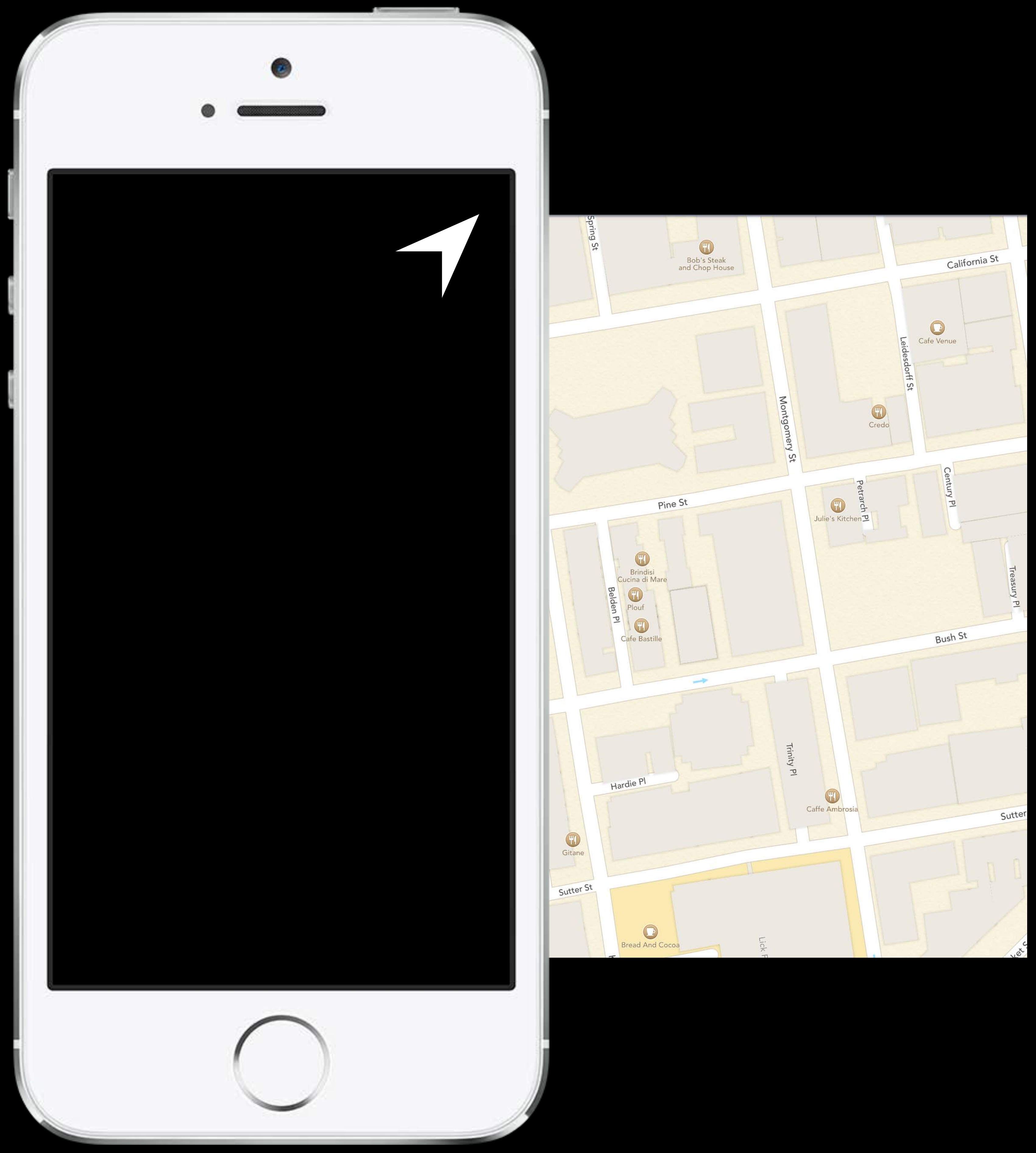
One Step Further

Run tracking app



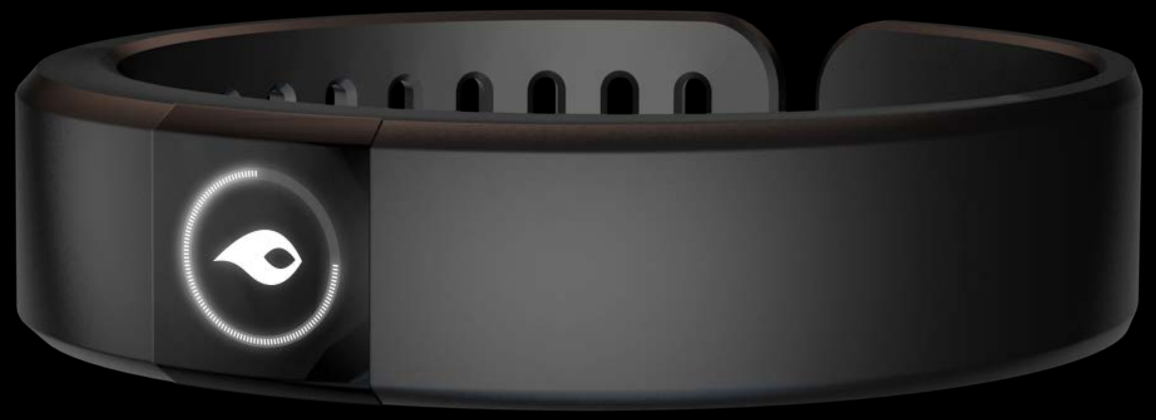
One Step Further

Run tracking app



One Step Further

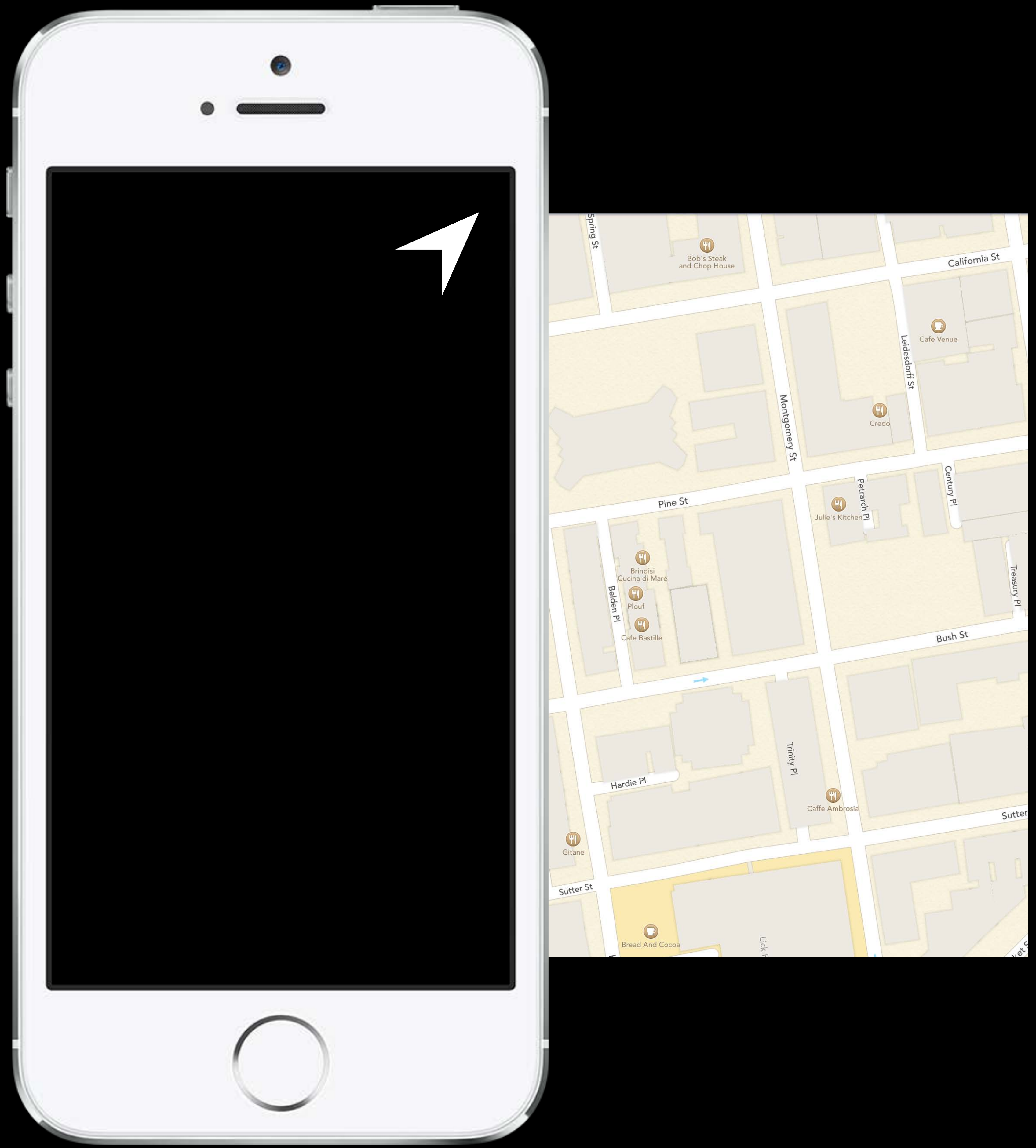
Run tracking app



Fitness Data

Fitness Data

Fitness Data



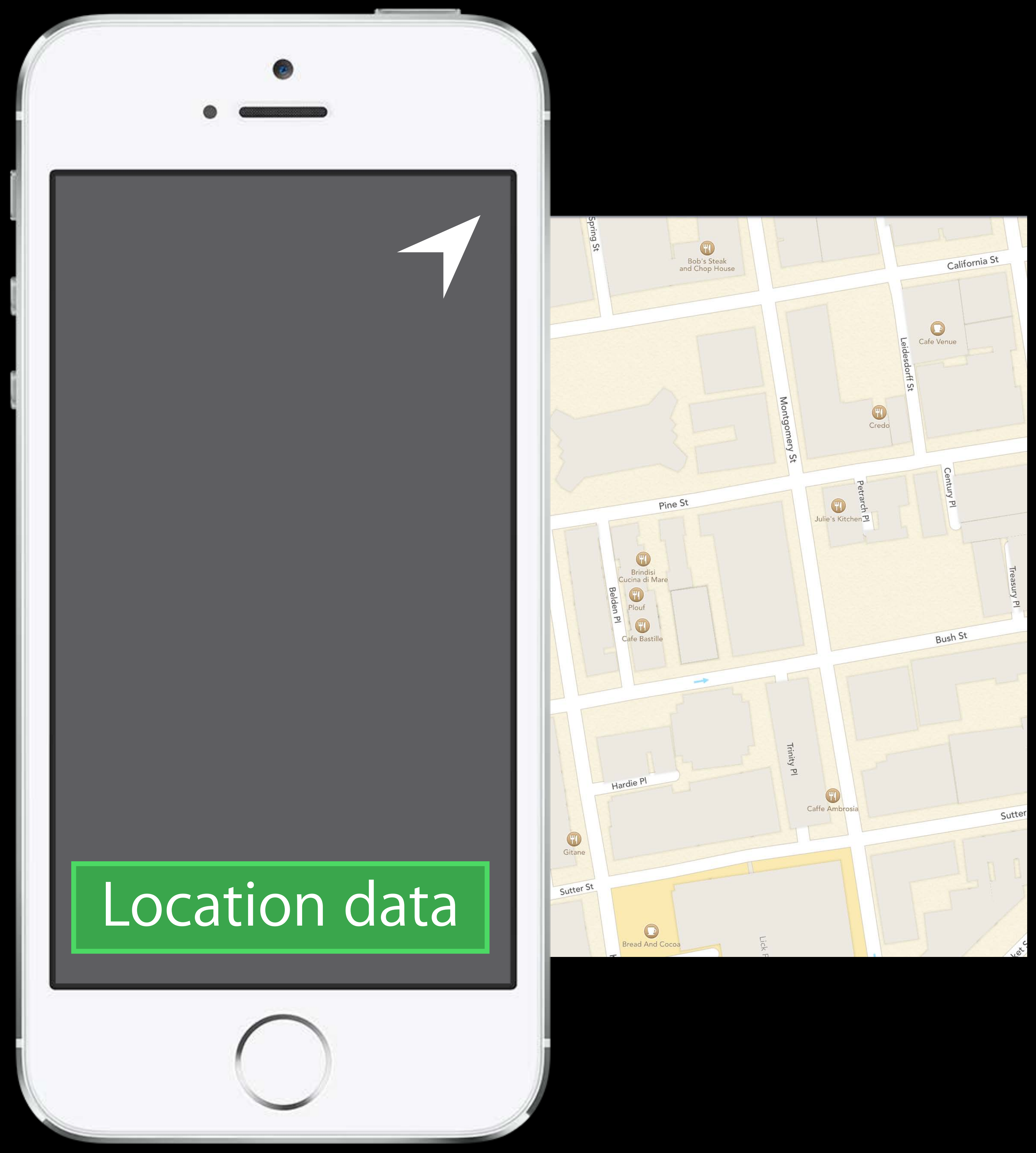
One Step Further

Run tracking app

Fitness Data

Fitness Data

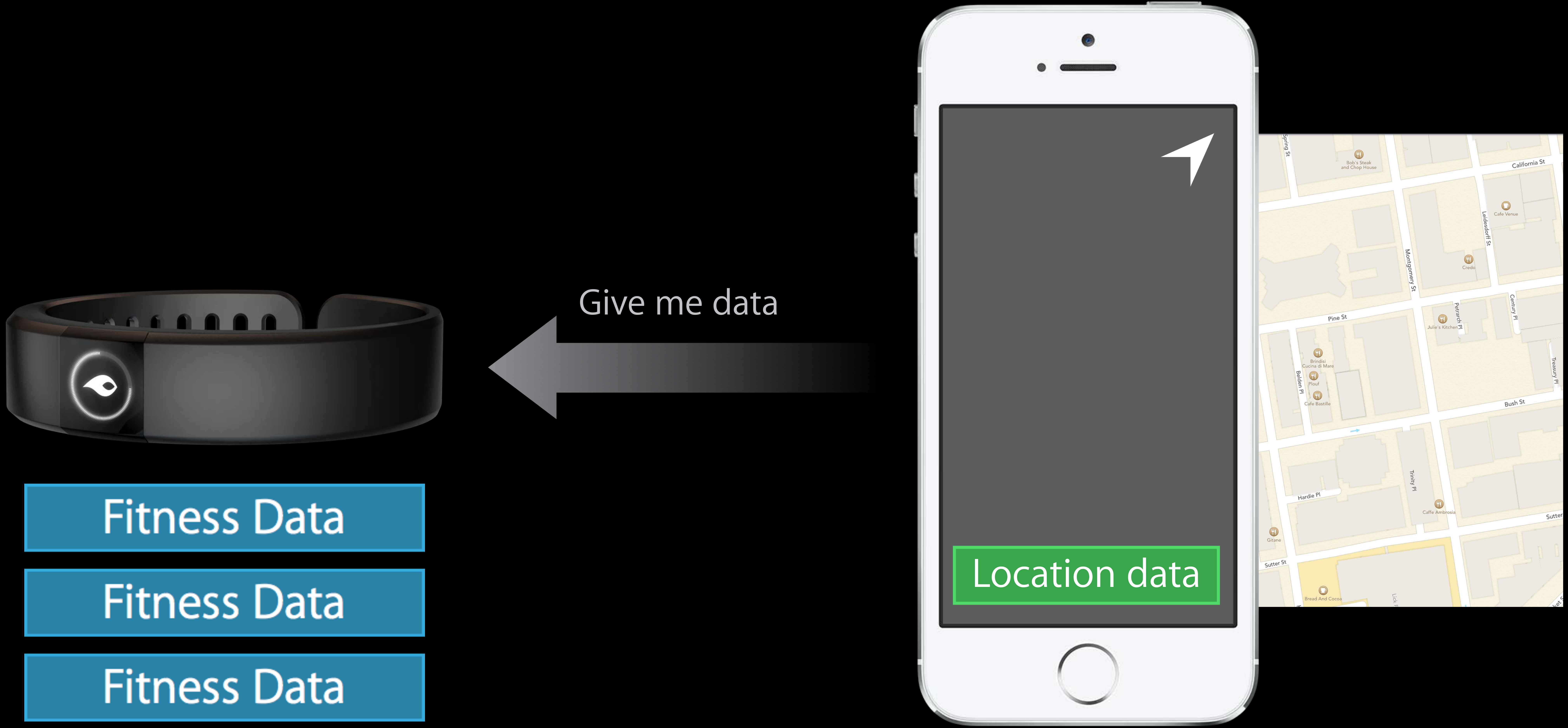
Fitness Data



Location data

One Step Further

Run tracking app



Give me data

Fitness Data

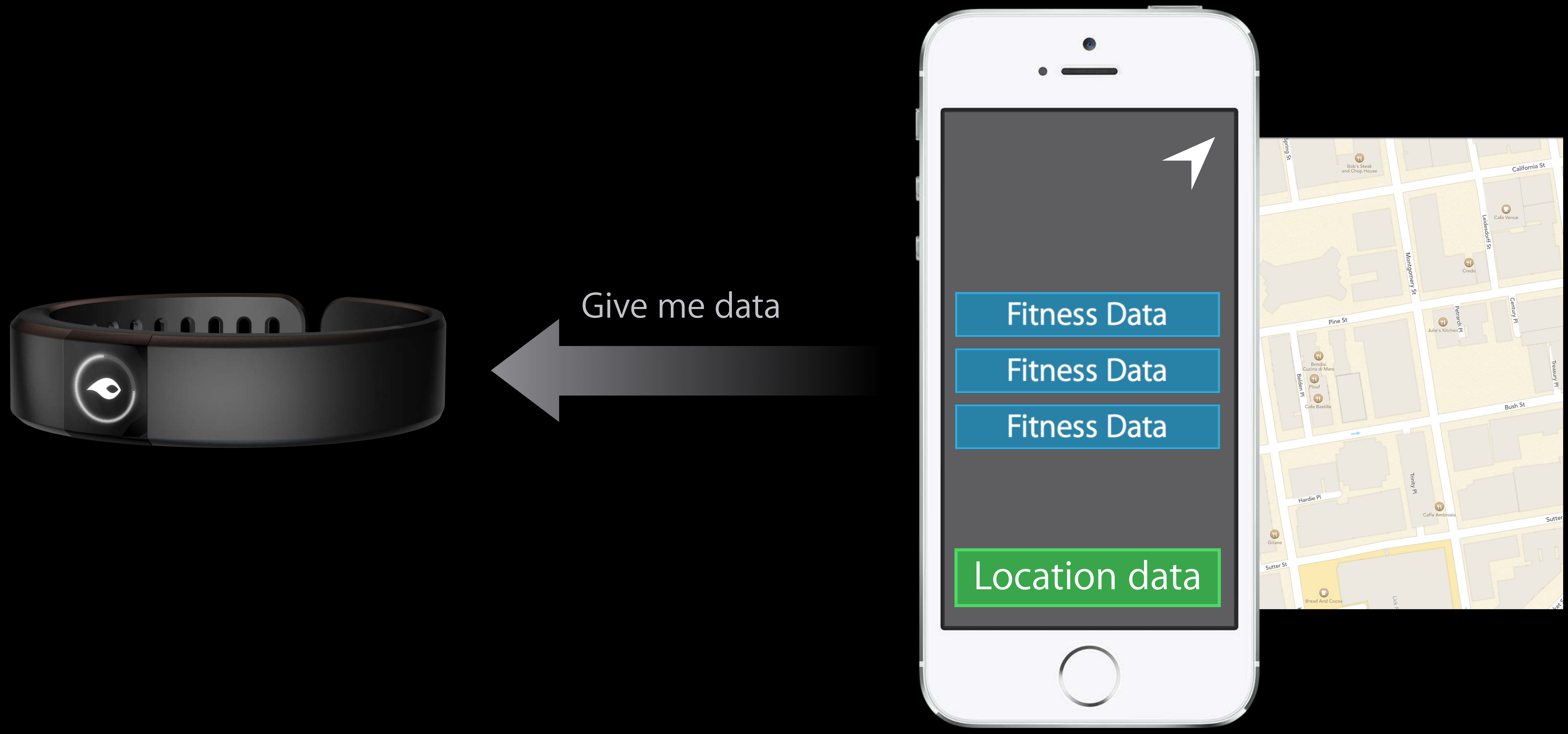
Fitness Data

Fitness Data

Location data

One Step Further

Run tracking app



Give me data

Fitness Data

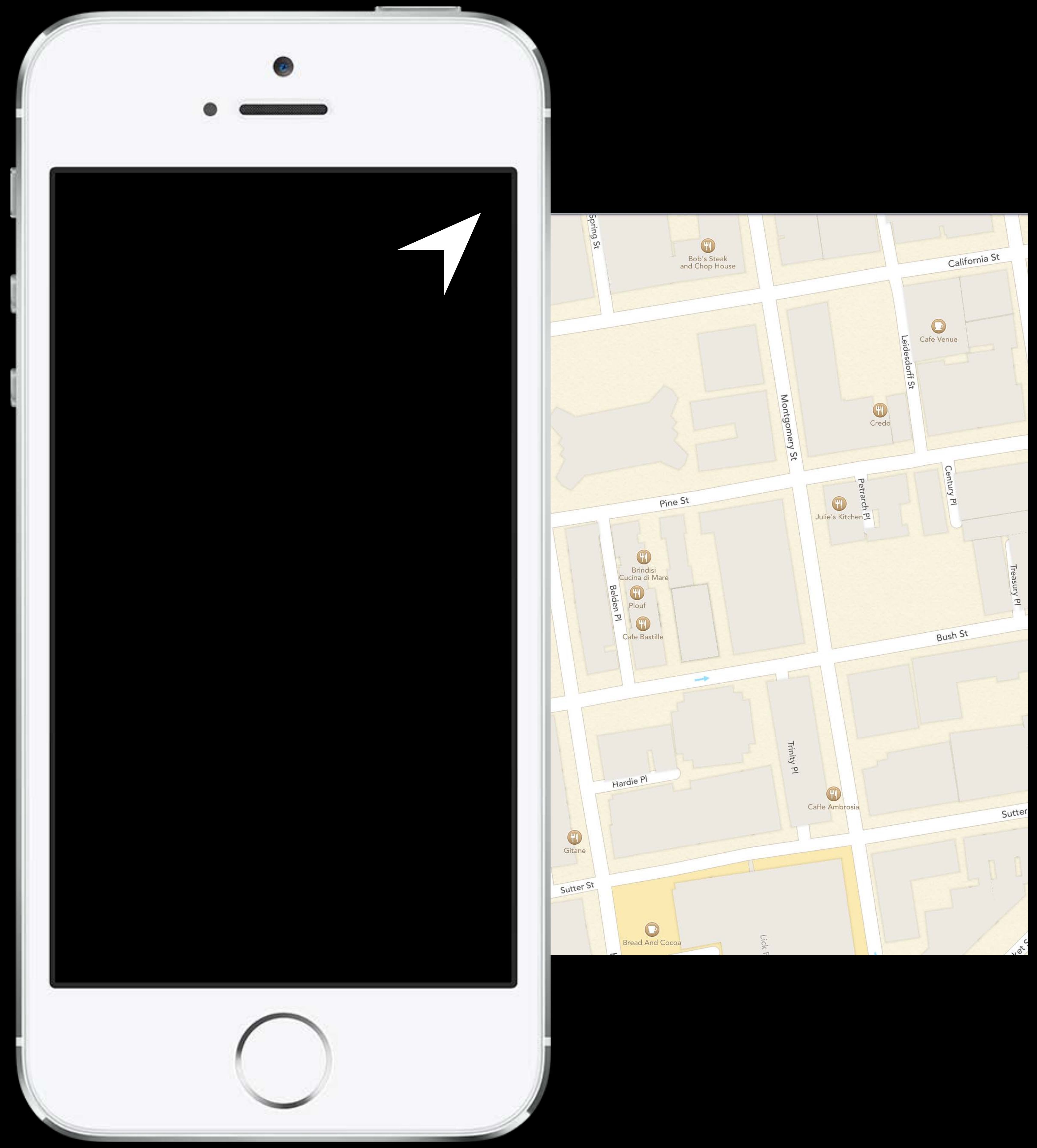
Fitness Data

Fitness Data

Location data

One Step Further

Run tracking app



Coalesce work
Cut down wakes

Final Thoughts

"With great power comes great responsibility…"

Summary

Your turn

Summary

Your turn

Battery life impacts user experience

Summary

Your turn

Battery life impacts user experience

Design your apps with the following in mind

Summary

Your turn

Battery life impacts user experience

Design your apps with the following in mind

- Do it never/less

Summary

Your turn

Battery life impacts user experience

Design your apps with the following in mind

- Do it never/less
- Do it more efficiently

Summary

Your turn

Battery life impacts user experience

Design your apps with the following in mind

- Do it never/less
- Do it more efficiently
- Do it at a better time

Summary

Your turn

Battery life impacts user experience

Design your apps with the following in mind

- Do it never/less
- Do it more efficiently
- Do it at a better time

Be a considerate background app

Summary

Your turn

Battery life impacts user experience

Design your apps with the following in mind

- Do it never/less
- Do it more efficiently
- Do it at a better time

Be a considerate background app

- Coalesce work

Summary

Your turn

- Battery life impacts user experience
- Design your apps with the following in mind
  - Do it never/less
  - Do it more efficiently
  - Do it at a better time
- Be a considerate background app
  - Coalesce work
  - Cutdown wakes

More Information

Paul Danbold

Core OS Technology Evangelist
danbold@apple.com

Jake Behrens

Apps Frameworks Evangelist
behrens@apple.com

Apple Developer Forums
http://devforums.apple.com

Related Sessions

🟠	Advanced Graphics and Animations for iOS Apps	Russian Hill	Tuesday 10:15AM
🔵	What's New in Core Location	Marina	Tuesday 2:00PM
🔵	What's New in Foundation Networking	Nob Hill	Tuesday 3:15PM
🟠	Improving Your App with Instruments	Marina	Tuesday 4:30PM
🔵	Writing Energy Efficient Code, Part I	Russian Hill	Wednesday 10:15AM
🔵	Power, Performance and Diagnostics: What's New in GCD and XPC	Russian Hill	Thursday 2:00PM

Labs

●	Power and Performance Lab	Core OS Lab B	Wednesday 2:00PM
●	Power and Performance Lab	Core OS Lab A	Thursday 3:15PM
●	Notifications Lab	Services Lab B	Wednesday 3:15PM
●	Instruments Lab	Tools Lab B	Thursday 9:00AM
●	Core Location Lab	Core OS Lab B	Thursday 12:45PM
●	Accessories and I/O Technologies Lab	Core OS Lab A	Thursday 12:45PM

WWDC14