Graphics and Games

#WWDC14

Motion Tracking with Core Motion Framework

Session 612

Andy Pham
Engineer

Sunny Chow
Engineer

1 What Is Motion Processing?

1. What Is Motion Processing?

2. How Can I Use it?

1. What Is Motion Processing?
2. How Can I Use it?
3. Deep Dive

1. What Is Motion Processing?
2. How Can I Use it?
3. Deep Dive
4. Let’s Code

What Is Motion Processing?

Motion Coprocessor

Motion Coprocessor

Always-On Motion Processing

+

Always-On Motion Processing

+

=

24/7 Motion
Awareness

Energy Efficient Motion Processing

Minimal cost to user

Energy Efficient Motion Processing

Minimal cost to user

24-Hr

Motion Activity

Pedometer

Energy Efficient Motion Processing

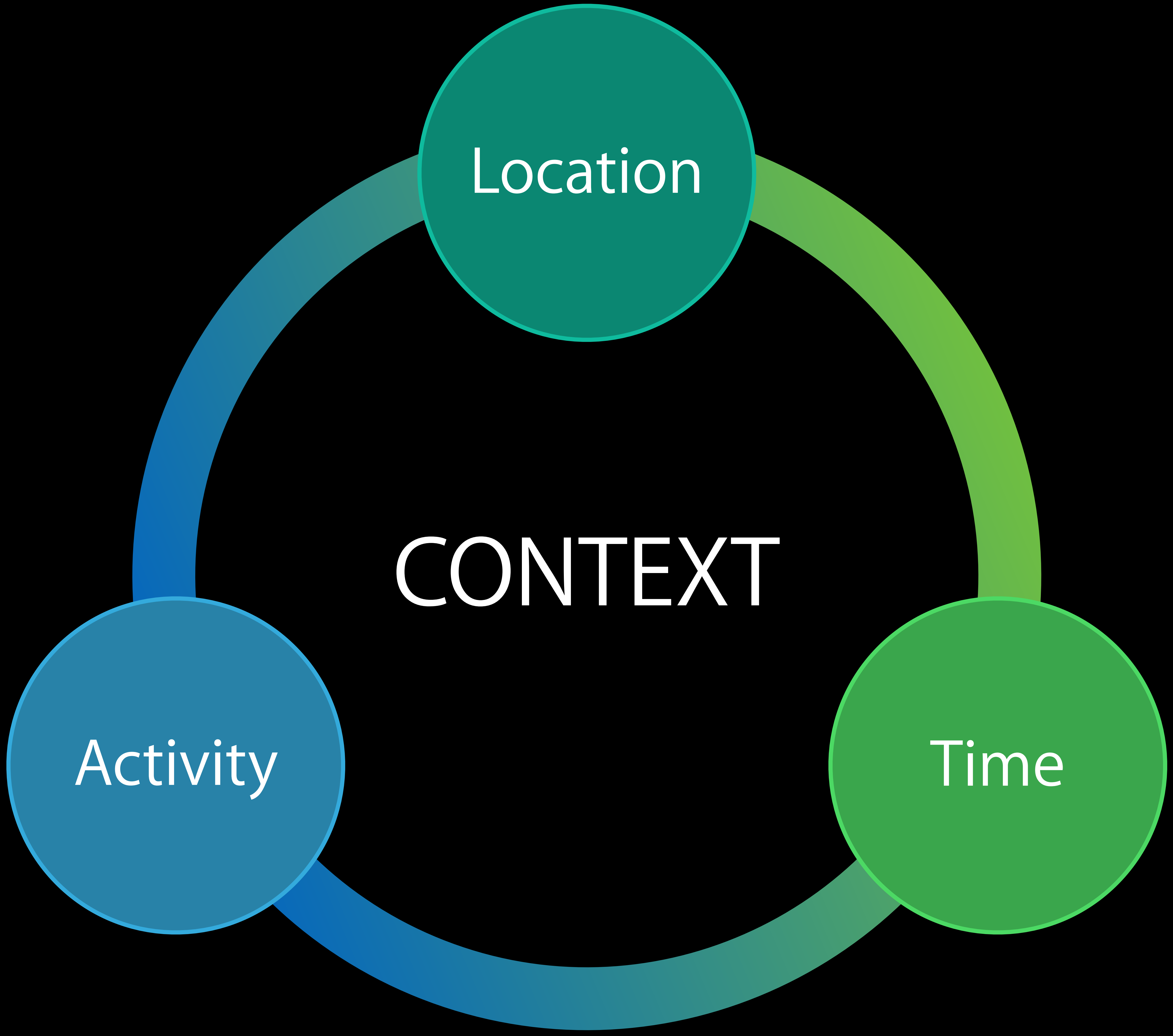
Minimal cost to user

24-Hr
Motion Activity
Pedometer

=

3-Min
FaceTime Call

Mobile Context



Motion Activity

Walking

Motion Activity

Walking

Performance is fairly insensitive to location

Motion Activity

Walking

Performance is fairly insensitive to location

- Detection can be suppressed when device is in hand

Motion Activity

Walking

Performance is fairly insensitive to location

- Detection can be suppressed when device is in hand

Relatively low latency

Motion Activity

Walking

Performance is fairly insensitive to location

- Detection can be suppressed when device is in hand

Relatively low latency

Very accurate, on average

Motion Activity

Walking

Performance is fairly insensitive to location

- Detection can be suppressed when device is in hand

Relatively low latency

Very accurate, on average

- Expect intermittent transitions into and out of walking state

Motion Activity

Running

Motion Activity

Running

Completely insensitive to location

Motion Activity

Running

Completely insensitive to location

Shortest latency

Motion Activity

Running

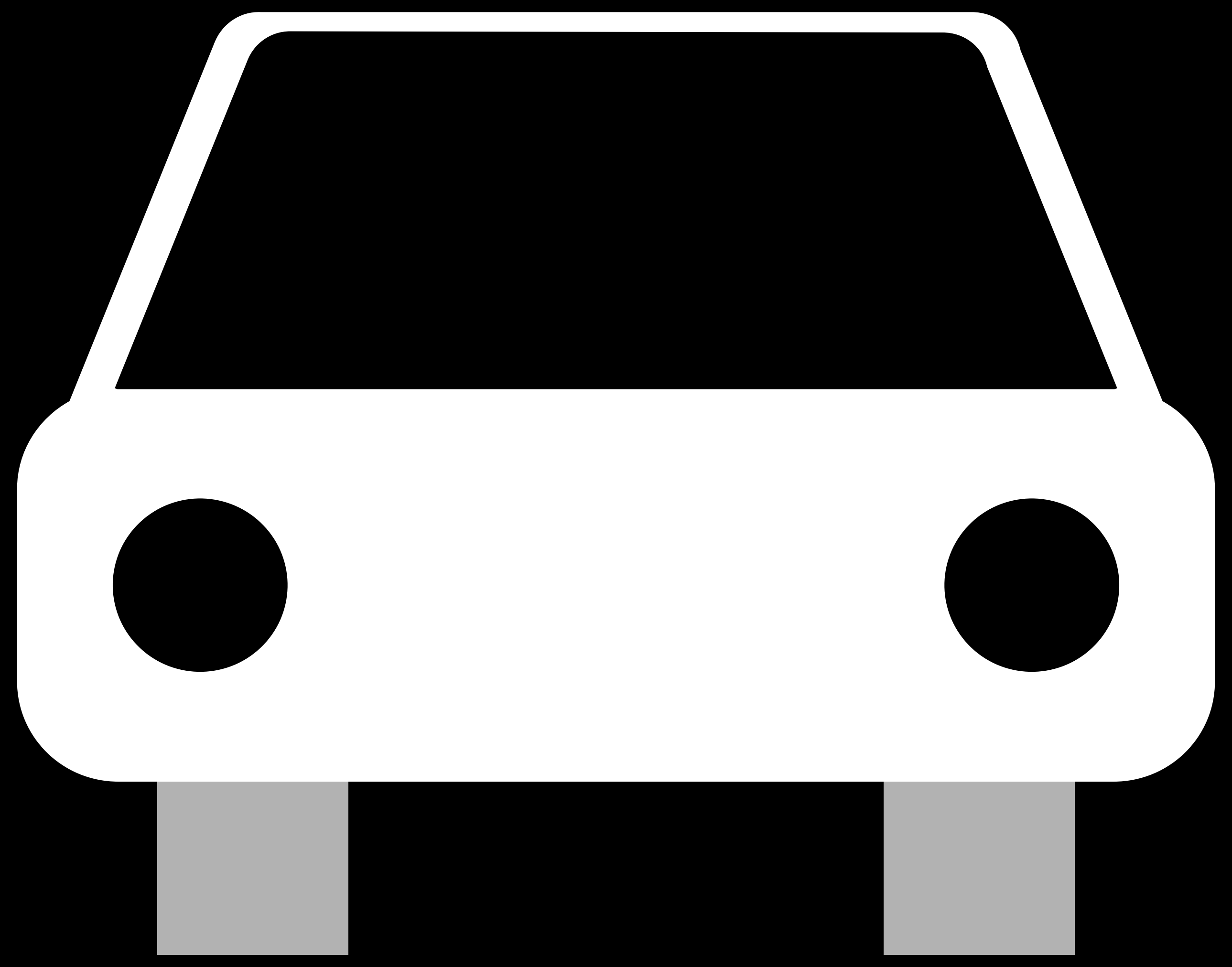
Completely insensitive to location

Shortest latency

Most accurate classification

Motion Activity

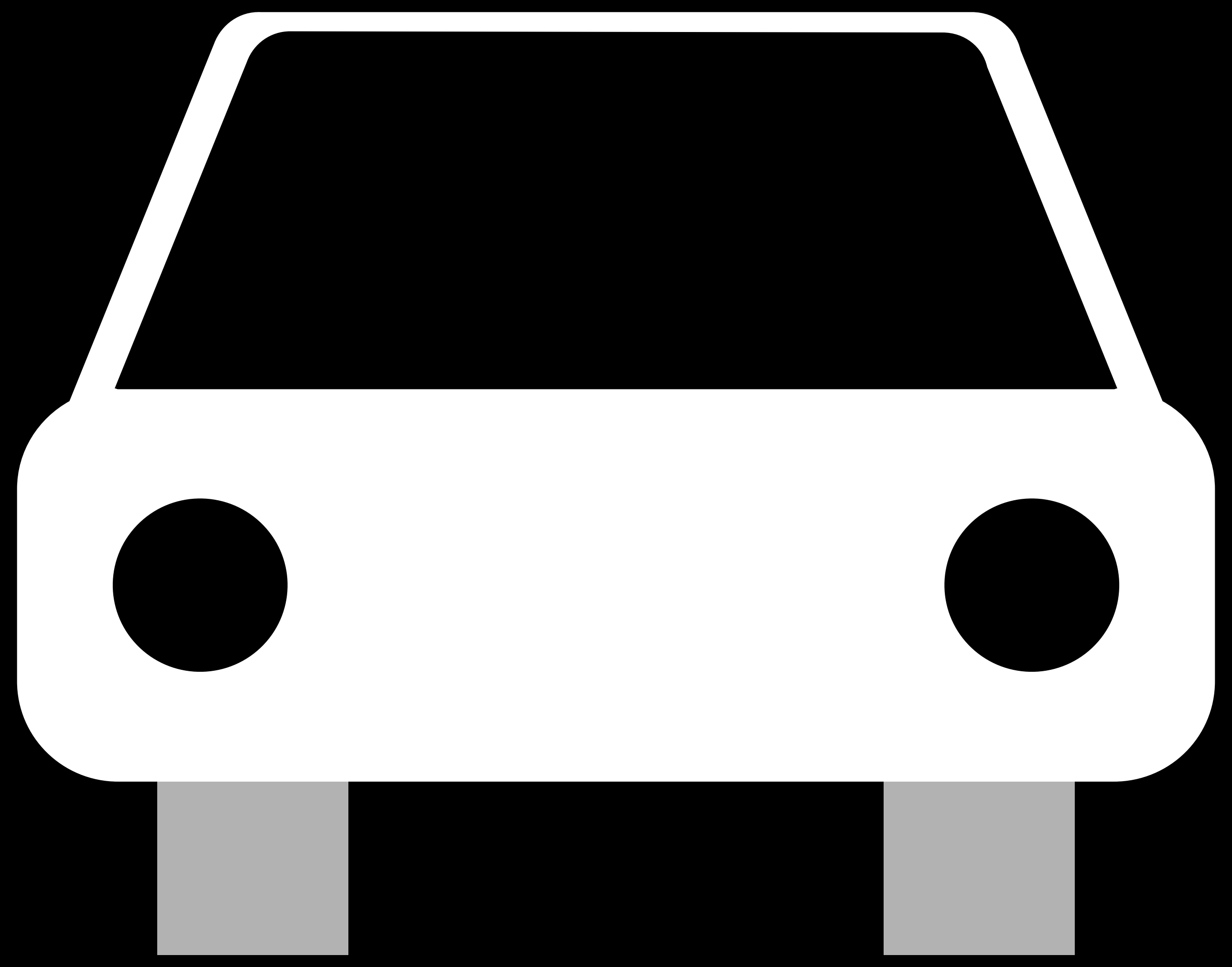
Automotive



Motion Activity

Automotive

Performance is sensitive to location

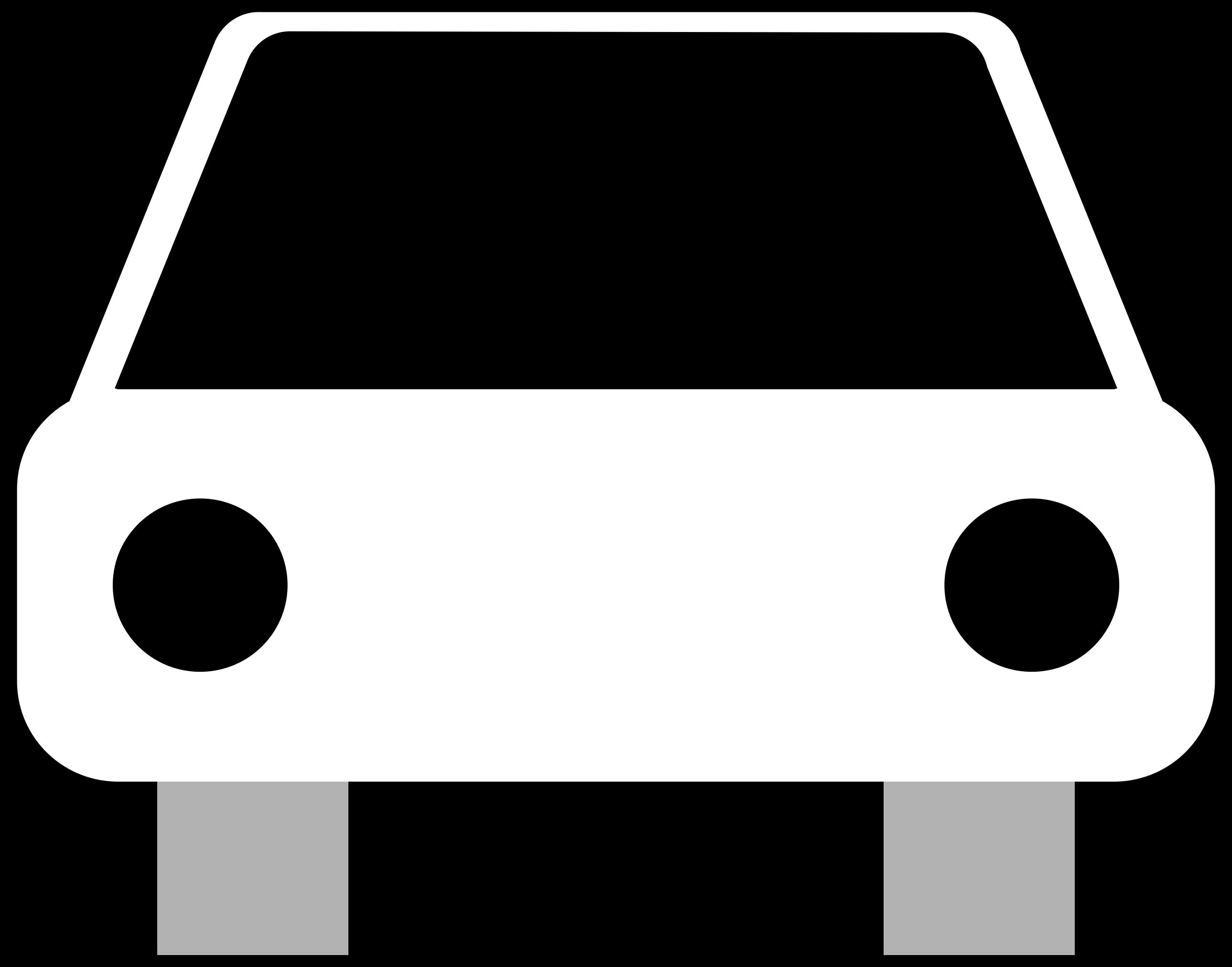


Motion Activity

Automotive

Performance is sensitive to location

- Works best if device is mounted, or placed in dash or in cup holder



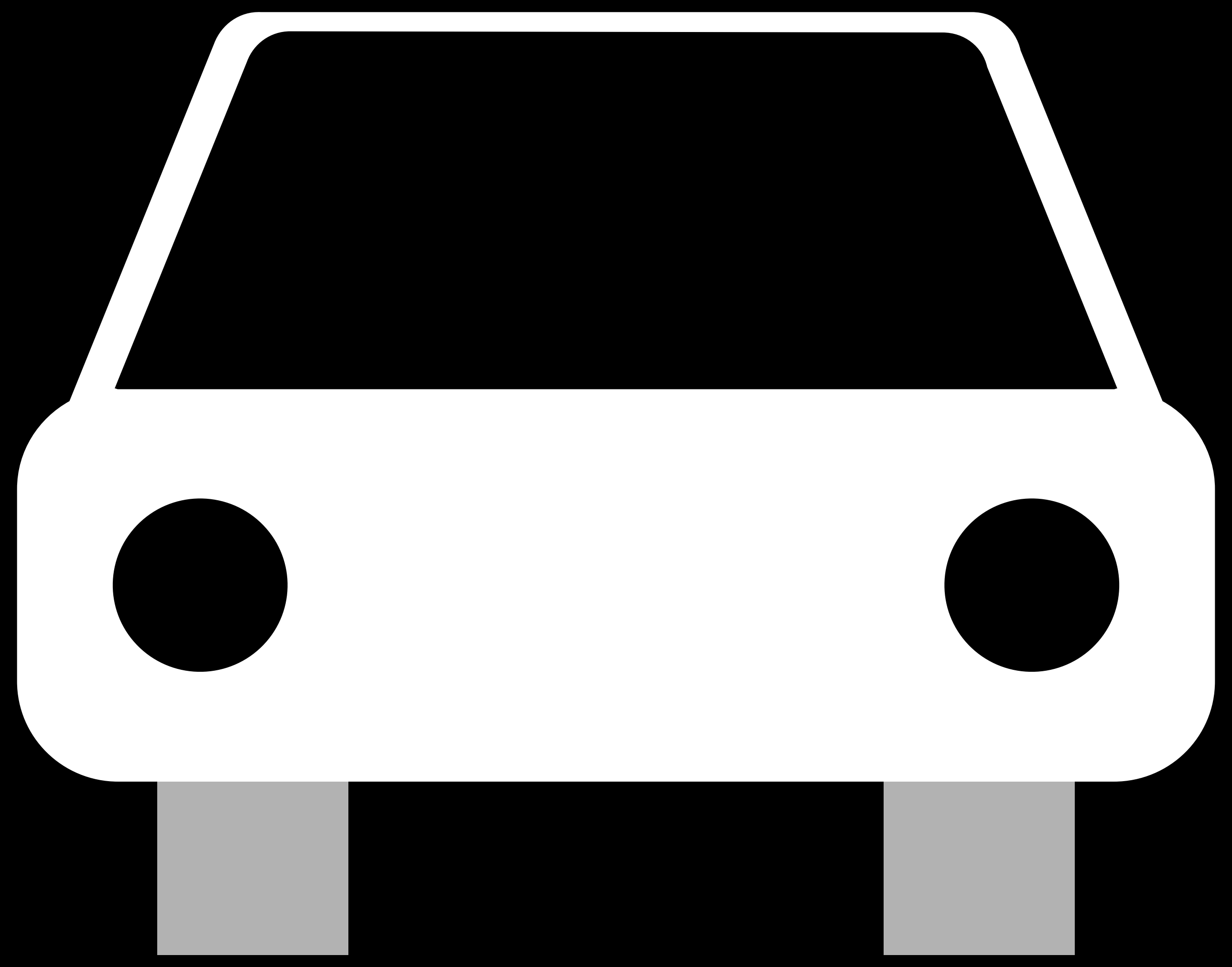
Motion Activity

Automotive

Performance is sensitive to location

- Works best if device is mounted, or placed in dash or in cup holder

Variable latency



Motion Activity

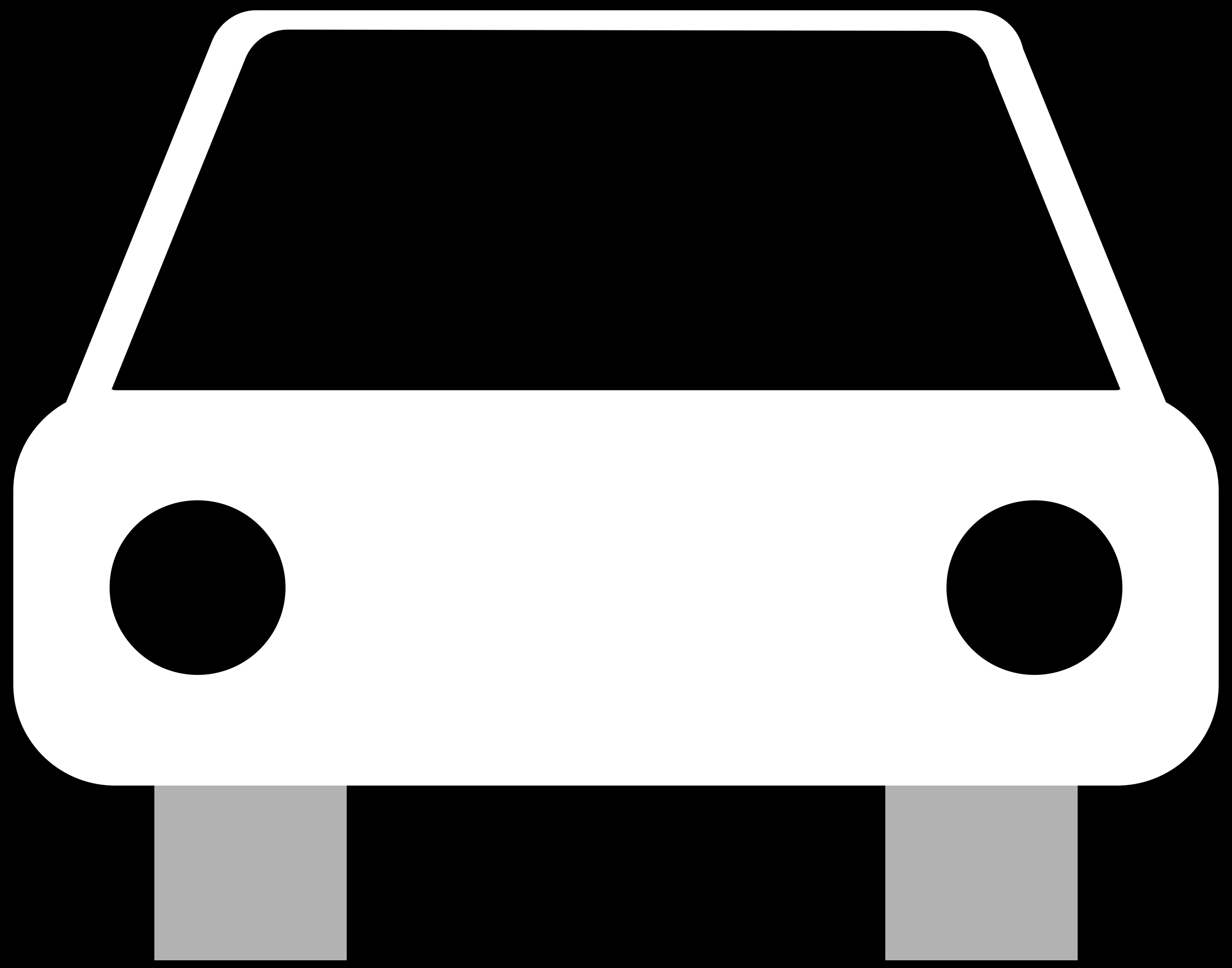
Automotive

Performance is sensitive to location

- Works best if device is mounted, or placed in dash or in cup holder

Variable latency

Relies on other information sources when available



Motion Activity

Cycling

Motion Activity

Cycling

Performance is very sensitive to location

Motion Activity

Cycling

Performance is very sensitive to location

- Works best if device is worn on upper arm

Motion Activity

Cycling

Performance is very sensitive to location

- Works best if device is worn on upper arm

Longest latency

Motion Activity

Cycling

Performance is very sensitive to location

- Works best if device is worn on upper arm

Longest latency

- Best for retrospective use cases

Health and Fitness

Step counting

- Provides measure of user’s activity level

Health and Fitness

Step counting

- Provides measure of user's activity level

Steps per day	Physical activity level
5,000	Sedentary
5,000–7,499	Underactive
7,500–9,999	Somewhat active
10,000–12,499	Active
12,500+	Highly active

Health and Fitness

Step counting

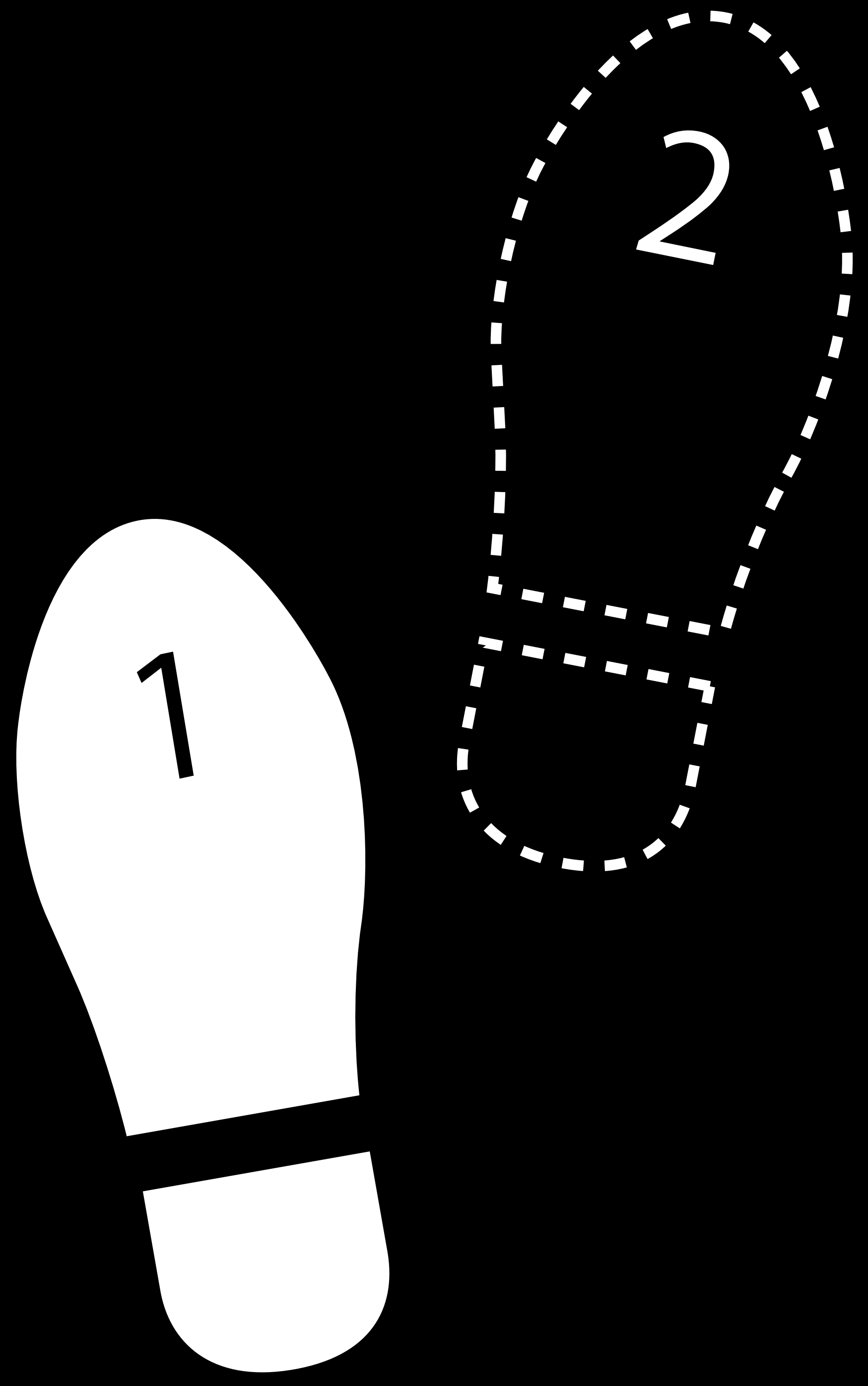
- Provides measure of user's activity level

Stride estimation

- Indicates intensity of user's workout

Pedometer

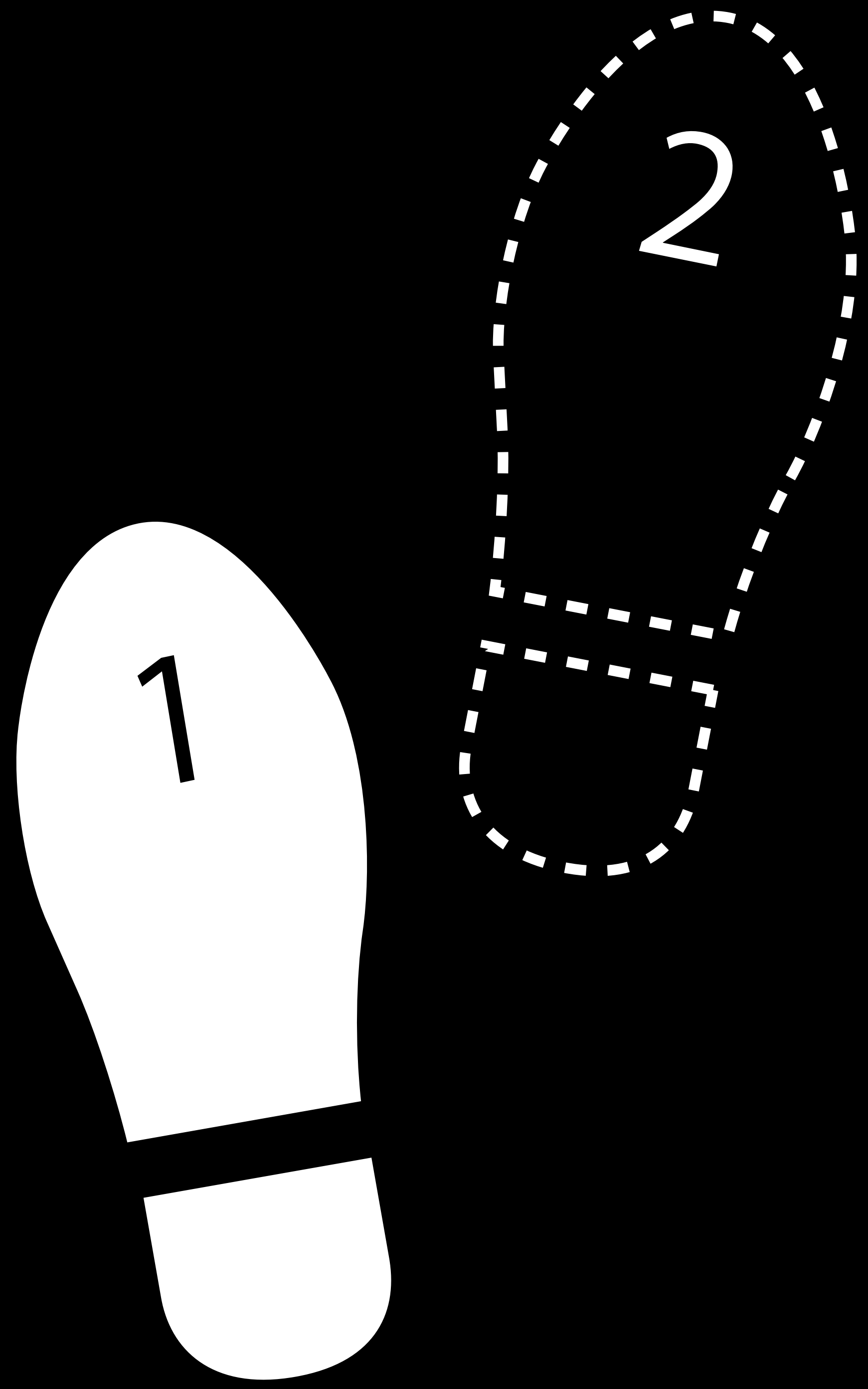
Step counting



Pedometer

Step counting

Consistent performance across body locations

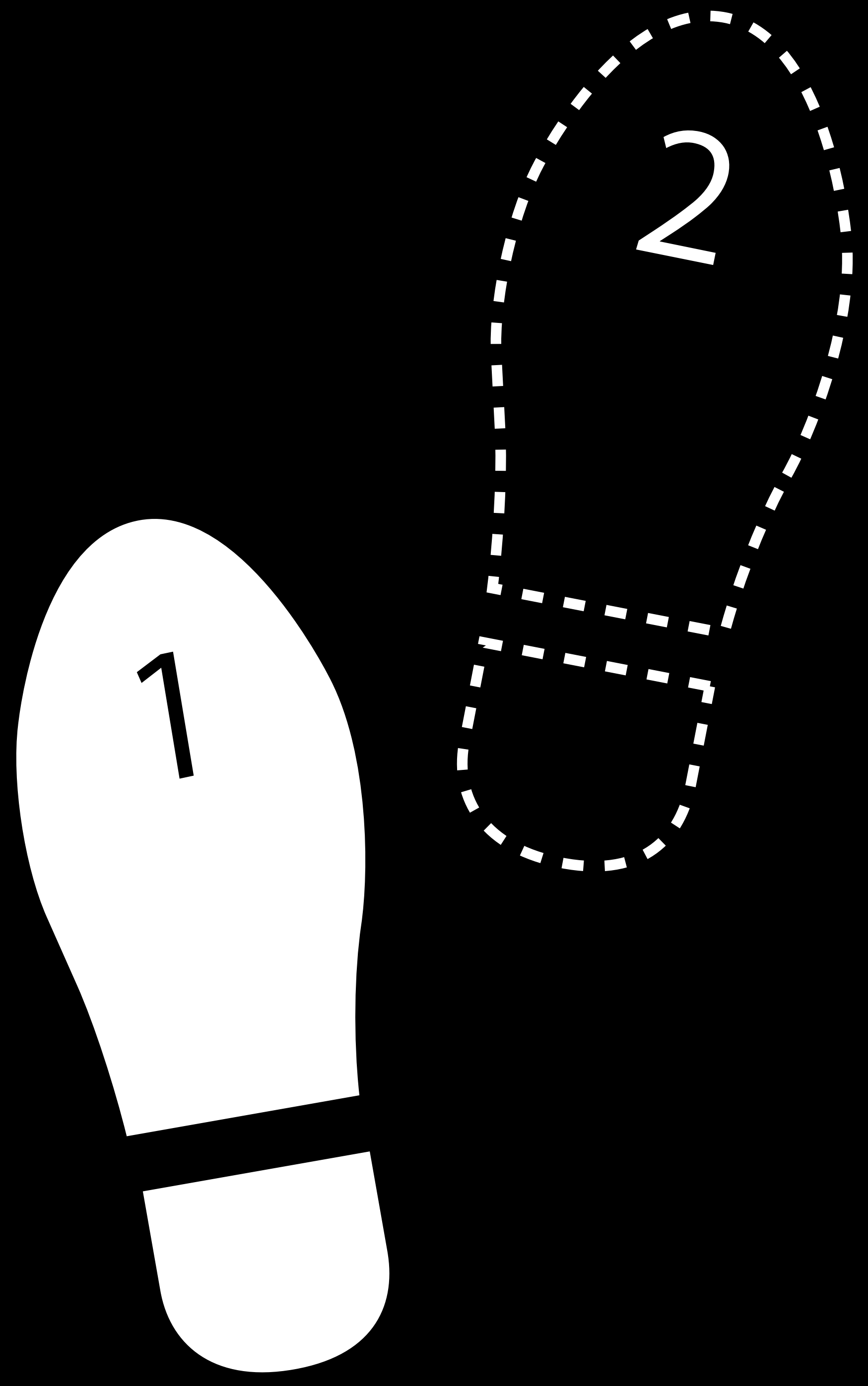


Pedometer

Step counting

Consistent performance across body locations

Extremely accurate



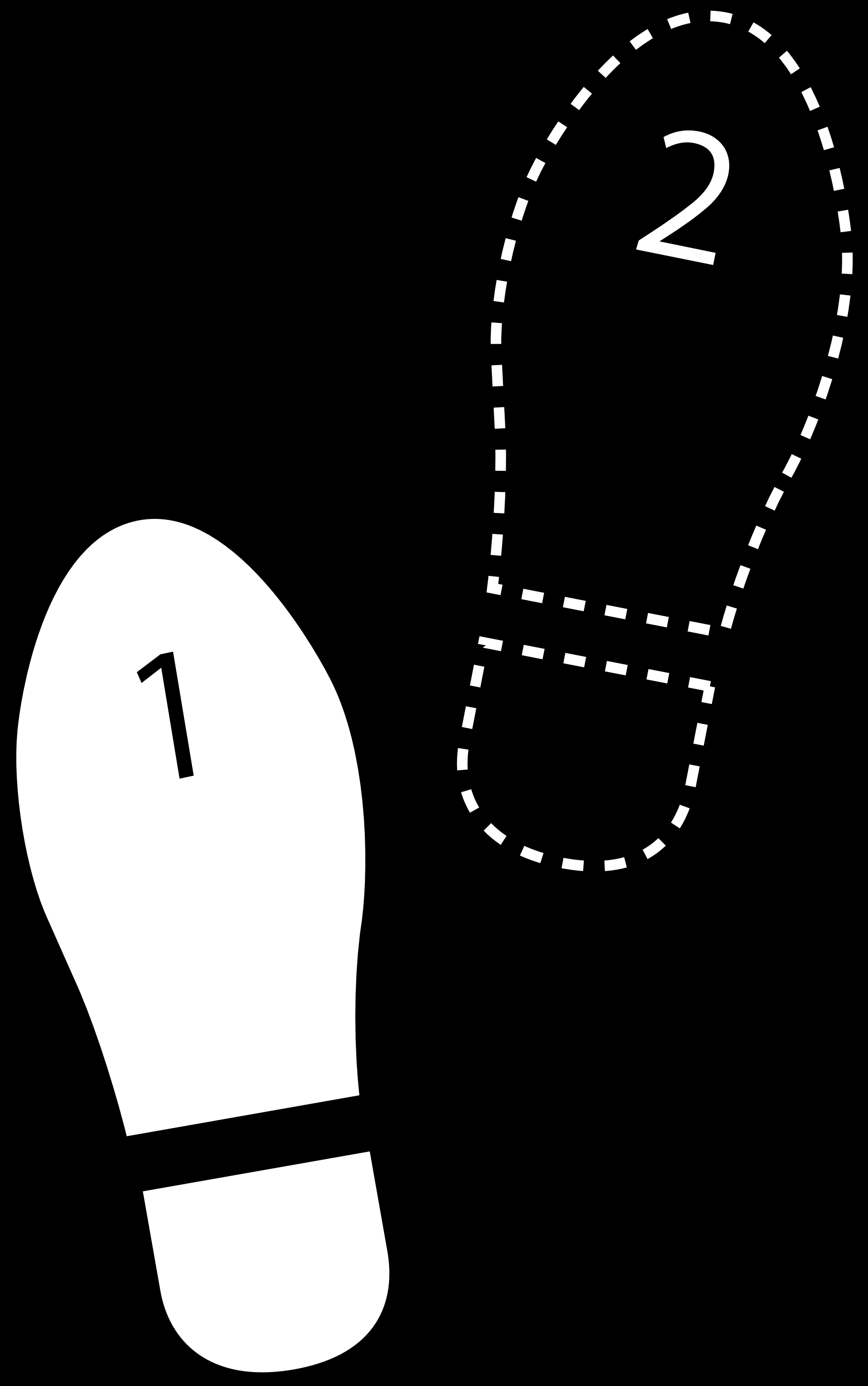
Pedometer

Step counting

Consistent performance across body locations

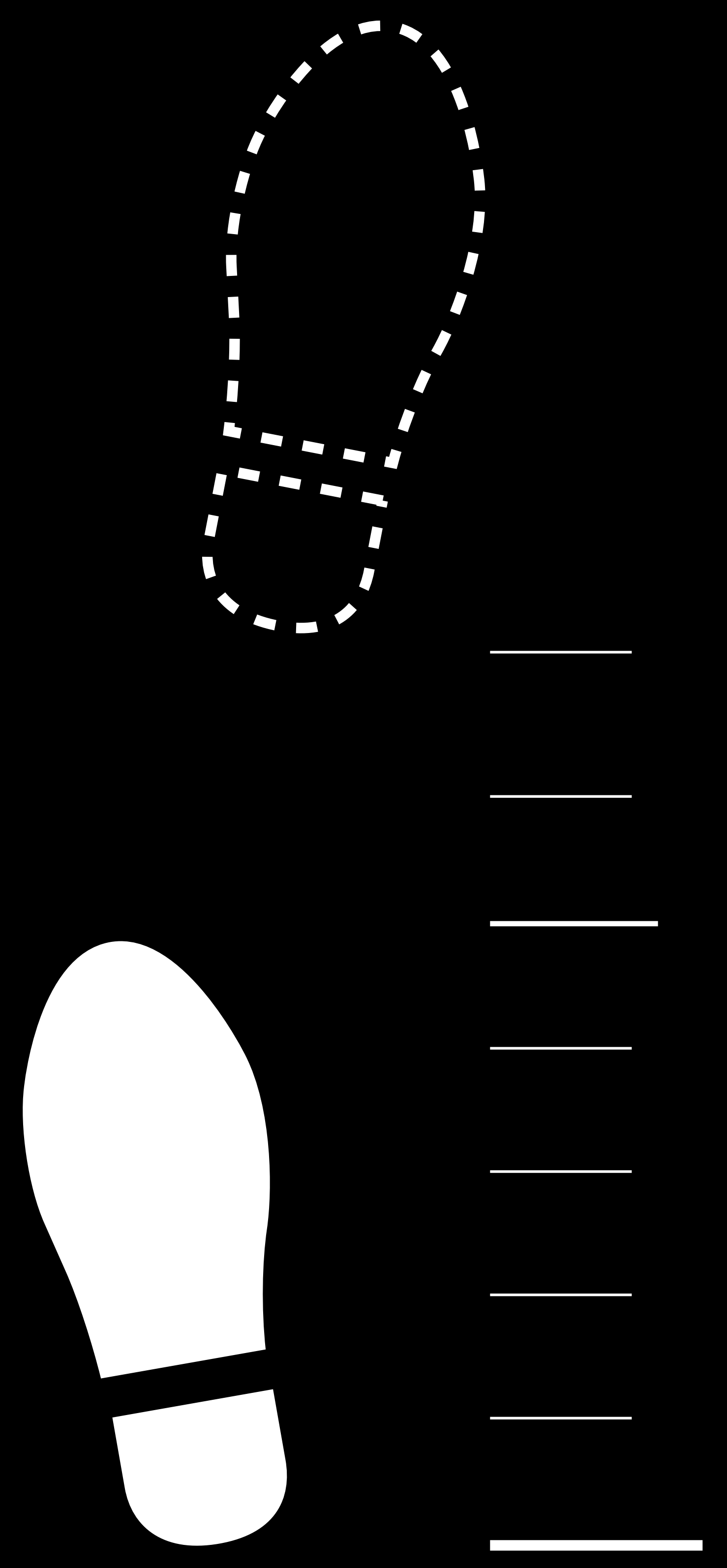
Extremely accurate

Robust to extraneous motions



Pedometer

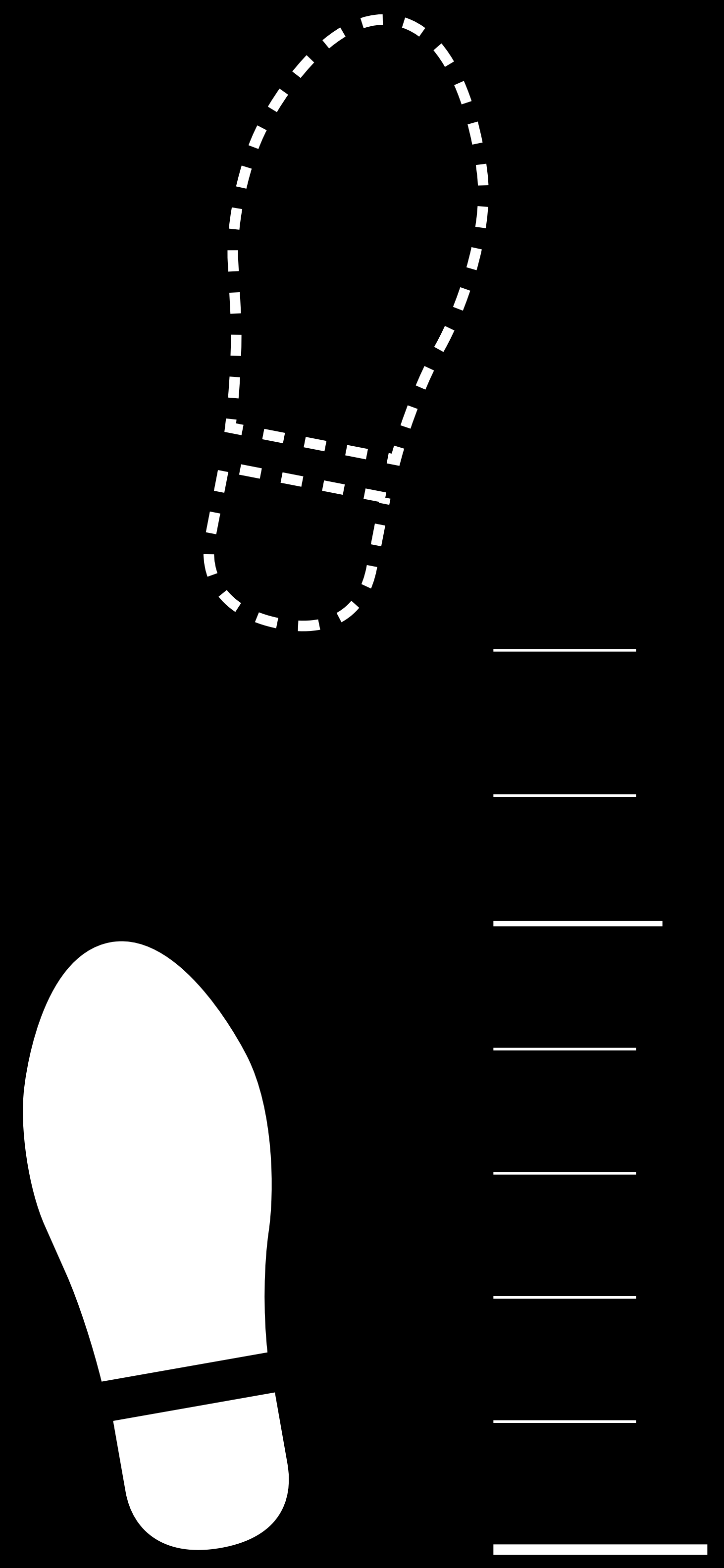
Stride estimation



Pedometer

Stride estimation

Consistent performance across body locations

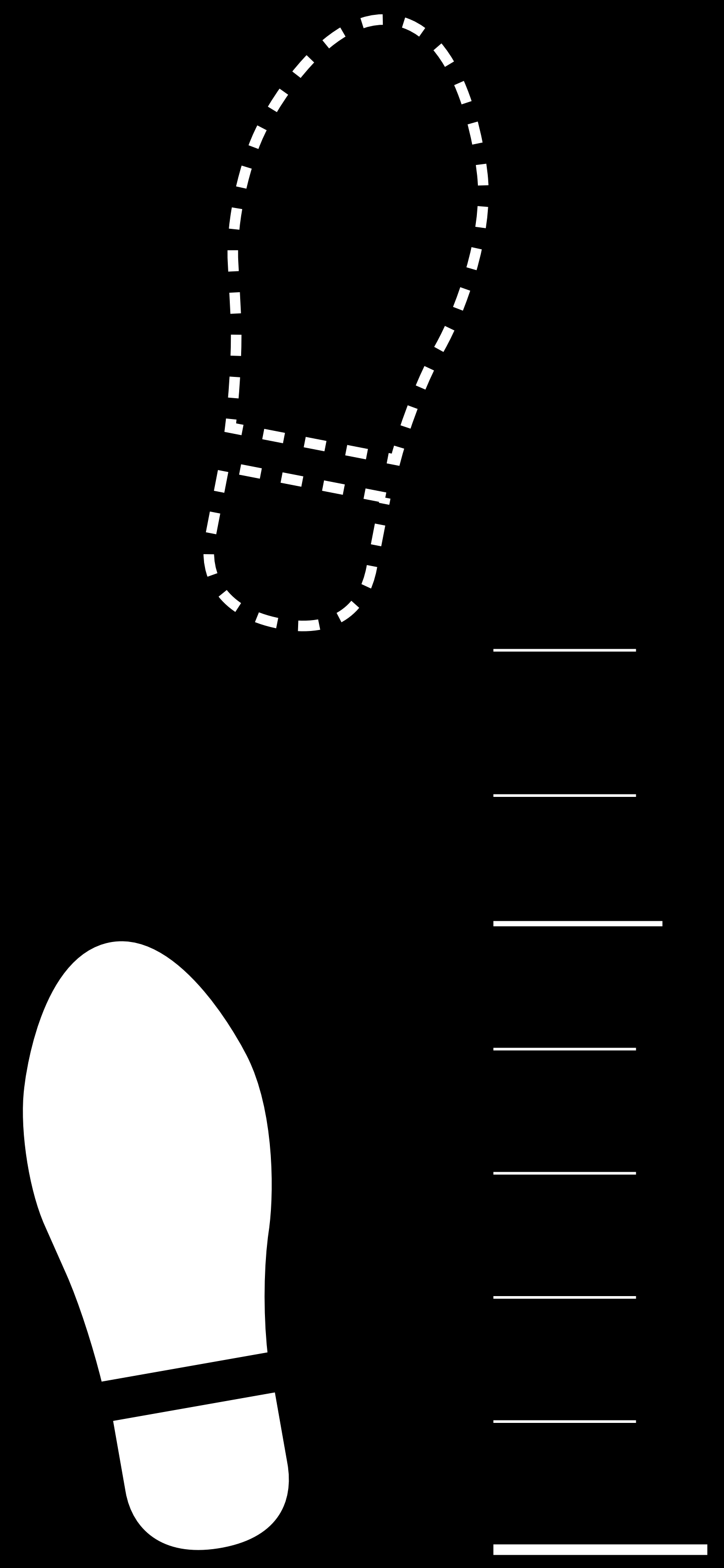


Pedometer

Stride estimation

Consistent performance across body locations

Consistent performance across pace



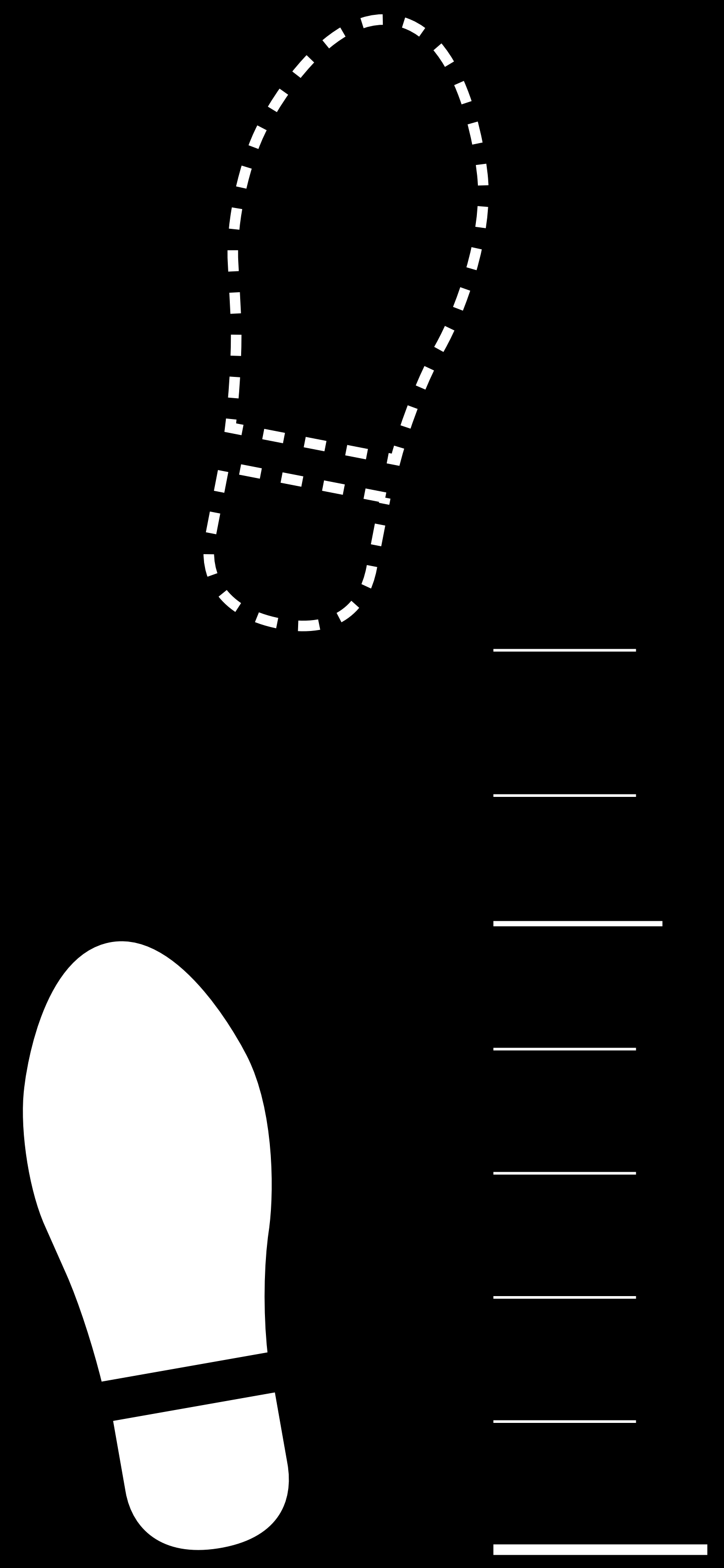
Pedometer

Stride estimation

Consistent performance across body locations

Consistent performance across pace

Extremely accurate



Pedometer

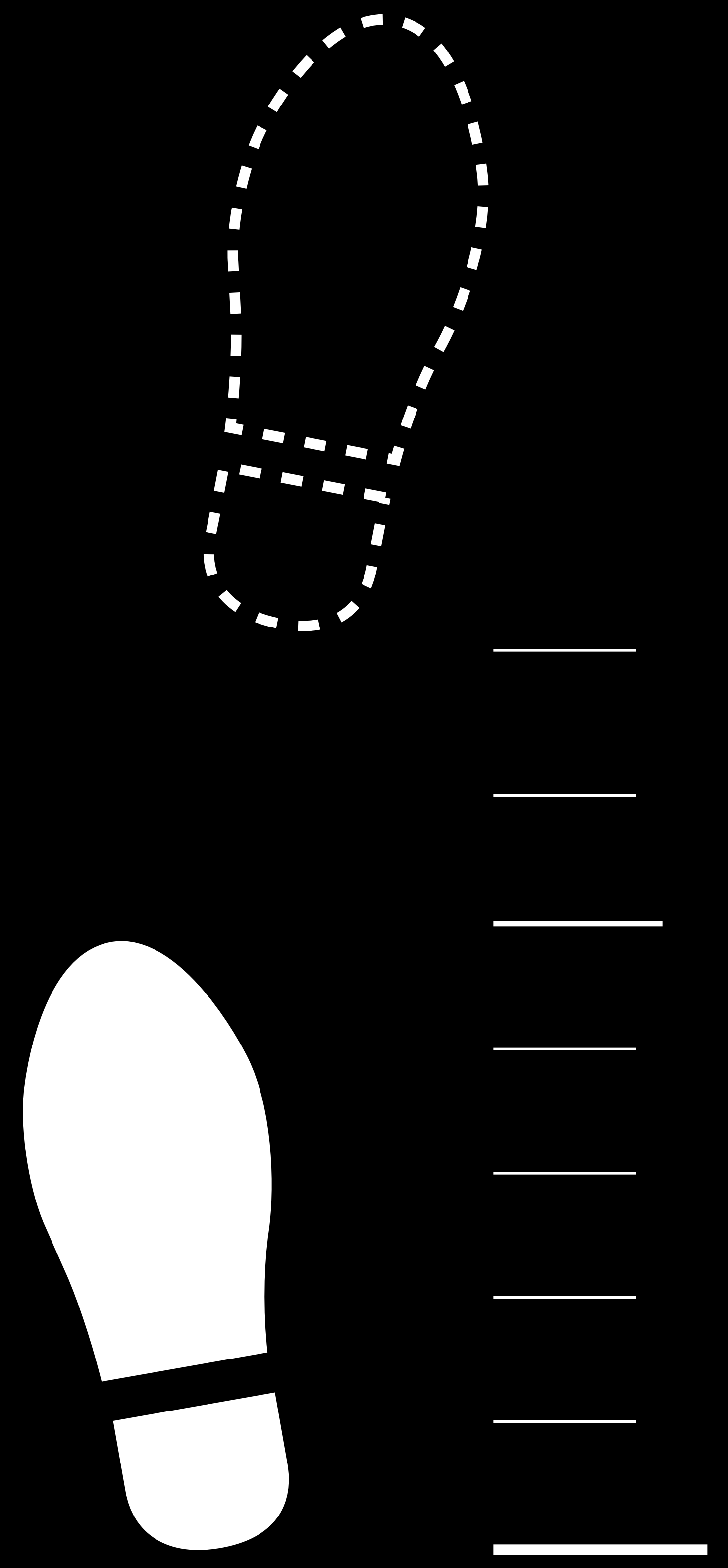
Stride estimation

Consistent performance across body locations

Consistent performance across pace

Extremely accurate

Adapts to the user over time



How Can I Use Motion Activity?

How Can I Receive Data?

Push vs. pull

How Can I Receive Data?

Push vs. pull

Push interface

How Can I Receive Data?

Push vs. pull

Push interface

- Apps receive event notifications with minimal latency

How Can I Receive Data?

Push vs. pull

Push interface

- Apps receive event notifications with minimal latency
    - Motion state transitions as they happen

How Can I Receive Data?

Push vs. pull

Push interface

- Apps receive event notifications with minimal latency
  - Motion state transitions as they happen
  - New step count and stride distance every 2.5s

How Can I Receive Data?

Push vs. pull

Push interface

- Apps receive event notifications with minimal latency
  - Motion state transitions as they happen
  - New step count and stride distance every 2.5s

Pull interface

How Can I Receive Data?

Push vs. pull

Push interface

- Apps receive event notifications with minimal latency
  - Motion state transitions as they happen
  - New step count and stride distance every 2.5s

Pull interface

- Apps query for motion information

How Can I Receive Data?

Push vs. pull

Push interface

- Apps receive event notifications with minimal latency
  - Motion state transitions as they happen
  - New step count and stride distance every 2.5s

Pull interface

- Apps query for motion information
  - Motion state transitions through a sequence of callbacks

How Can I Receive Data?

Push vs. pull

Push interface

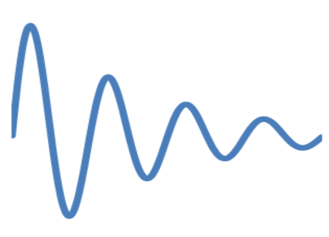
- Apps receive event notifications with minimal latency
  - Motion state transitions as they happen
  - New step count and stride distance every 2.5s

Pull interface

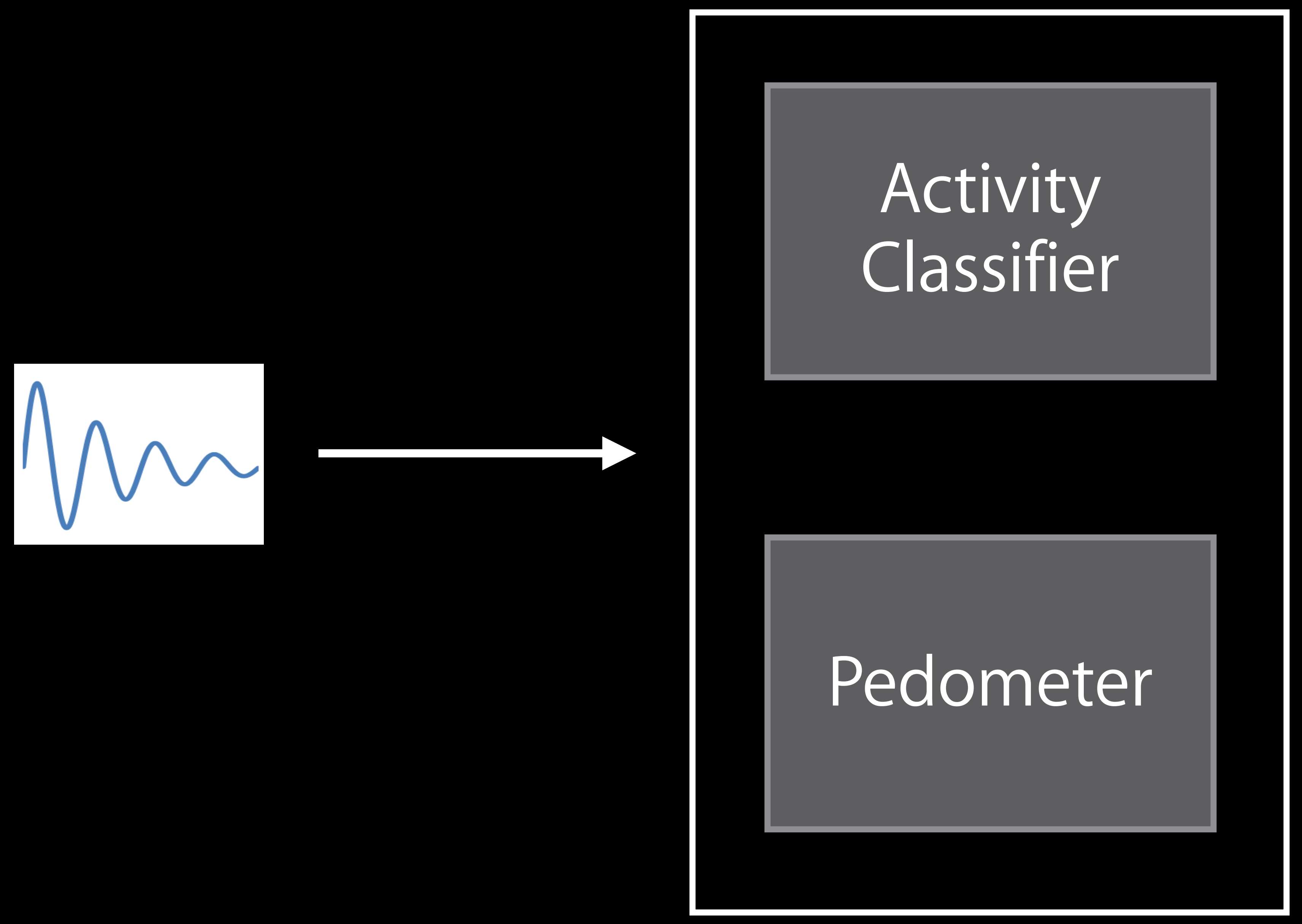
- Apps query for motion information
  - Motion state transitions through a sequence of callbacks
  - Single pedometer record containing cumulative step counts and distance

Motion Processing Architecture

Motion Processing Architecture



Motion Processing Architecture

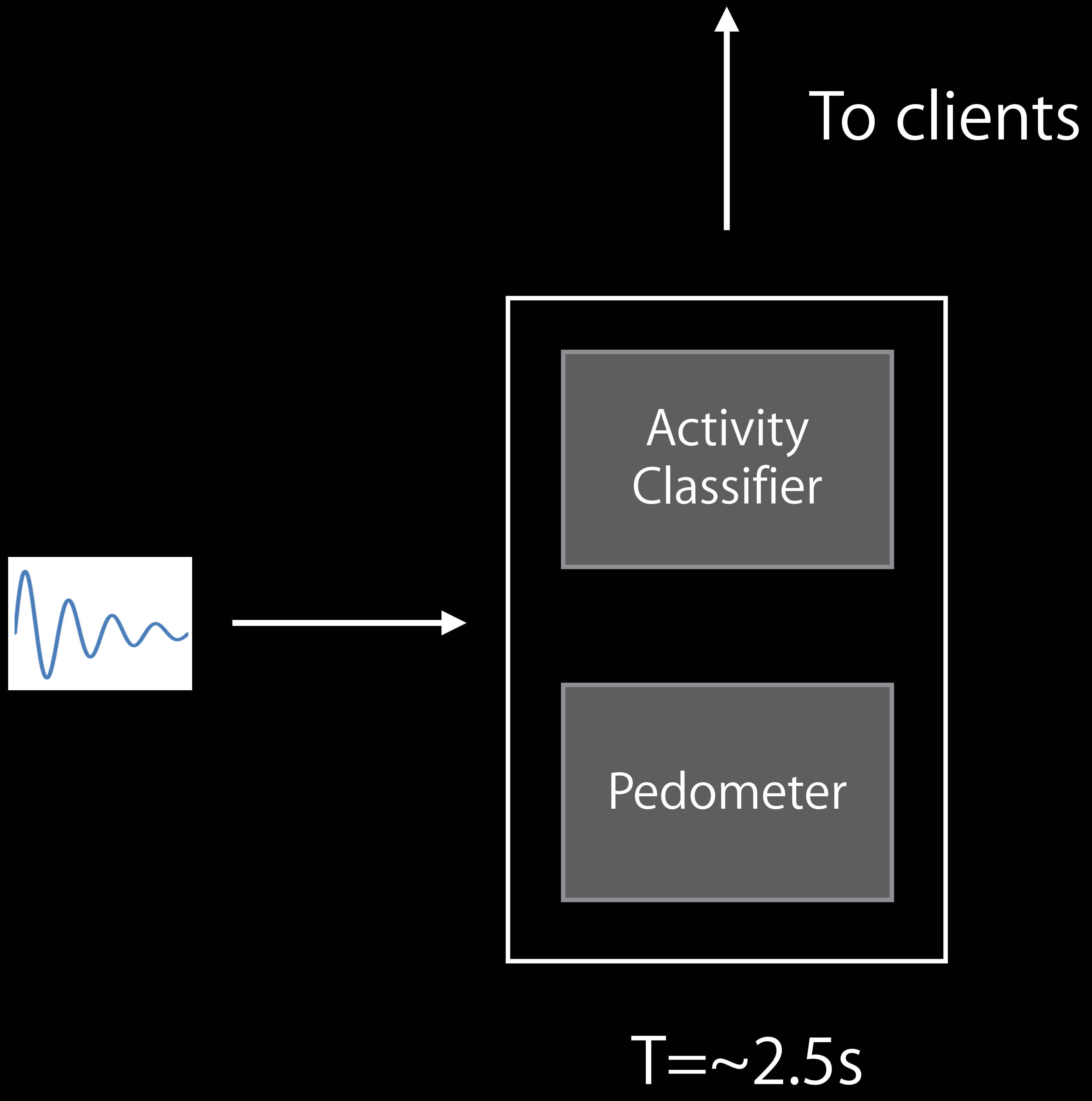


Activity Classifier

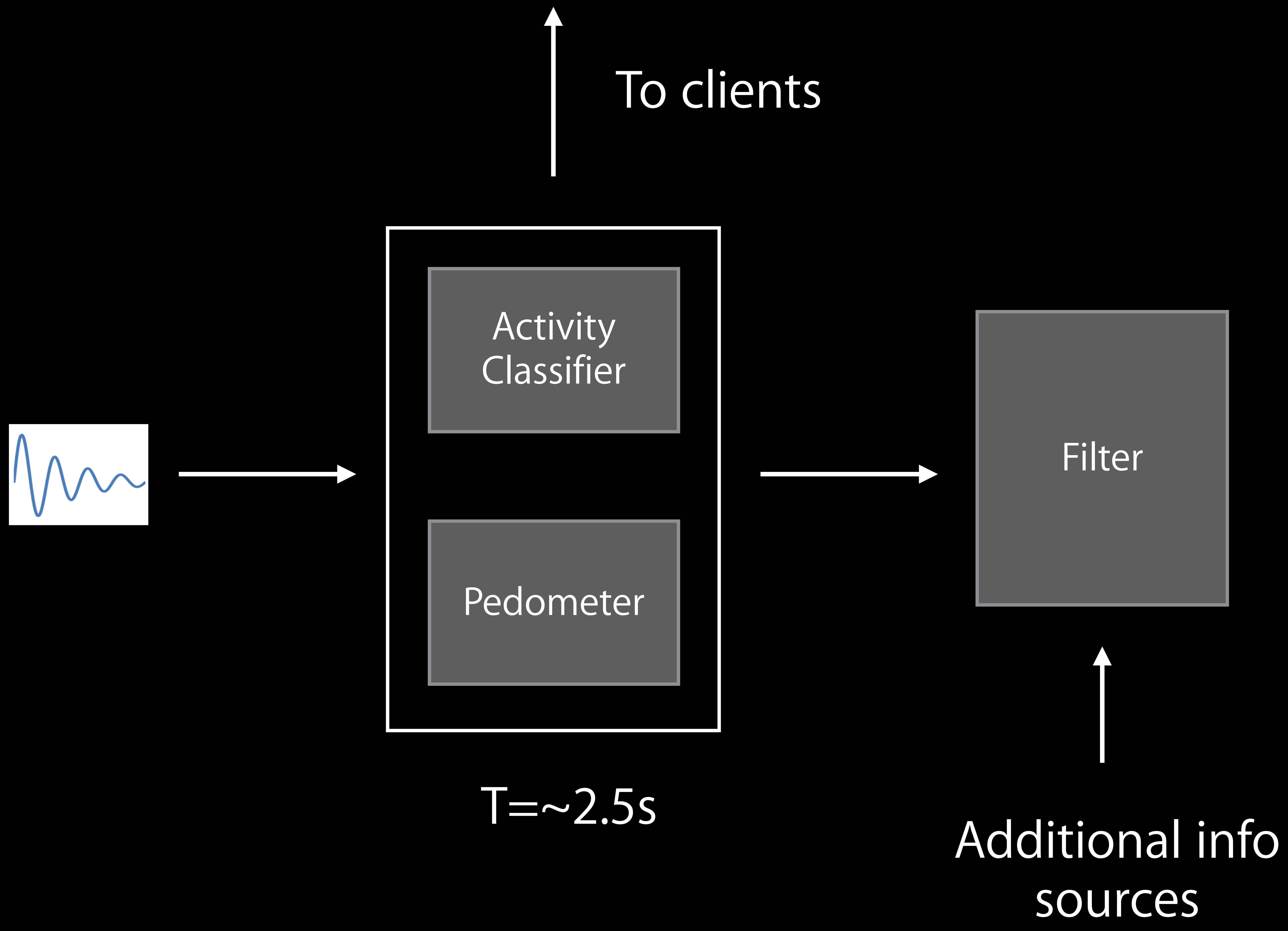
Pedometer

T=~2.5s

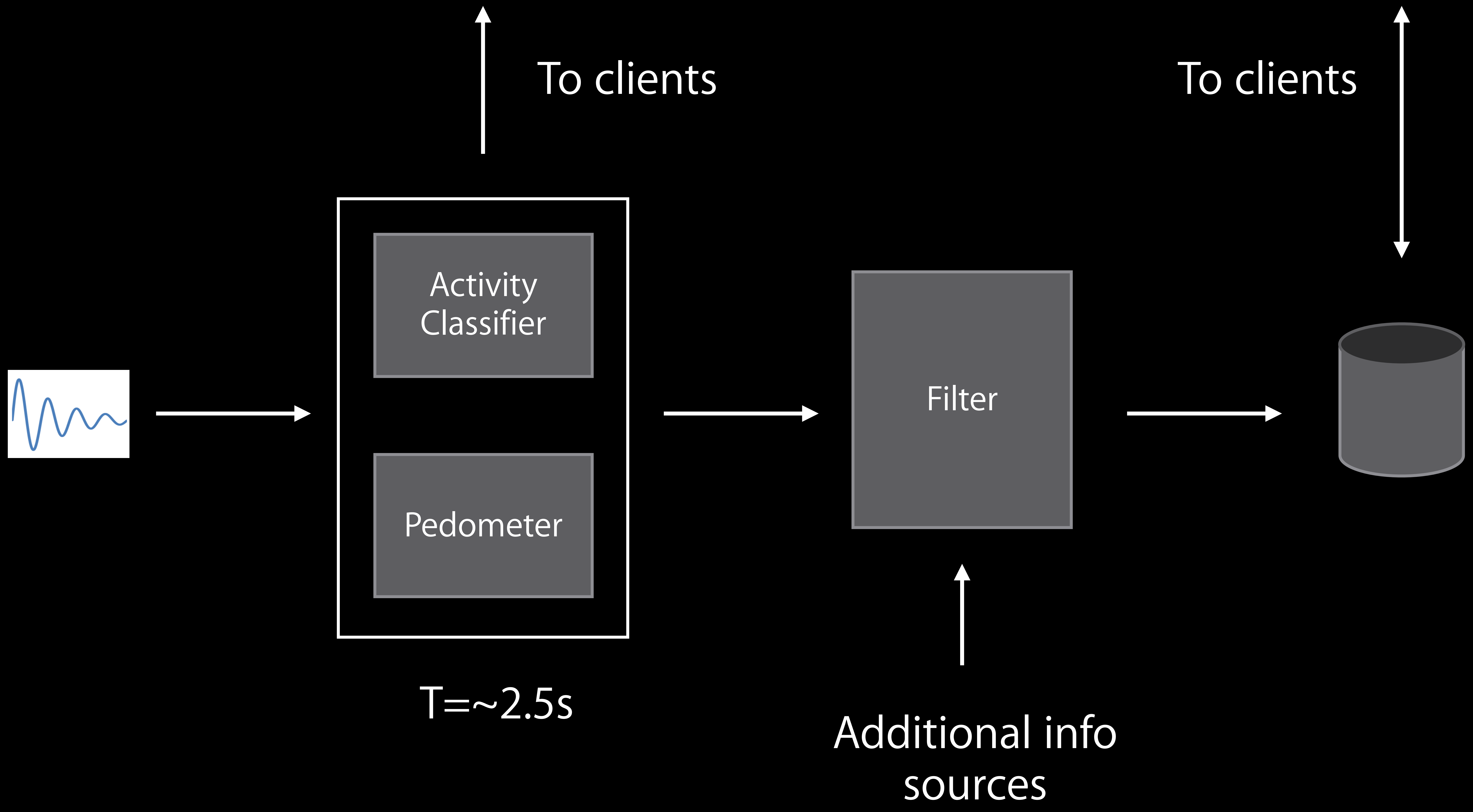
Motion Processing Architecture



Motion Processing Architecture



Motion Processing Architecture



Push Interface

Running app example

Push Interface

Running app example

What I would want in a running app

Push Interface

Running app example

What I would want in a running app

- Automatically change my experience throughout workout

Push Interface

Running app example

What I would want in a running app

- Automatically change my experience throughout workout
- Provide real-time feedback to me as I run

Push Interface

Running app example

What I would want in a running app

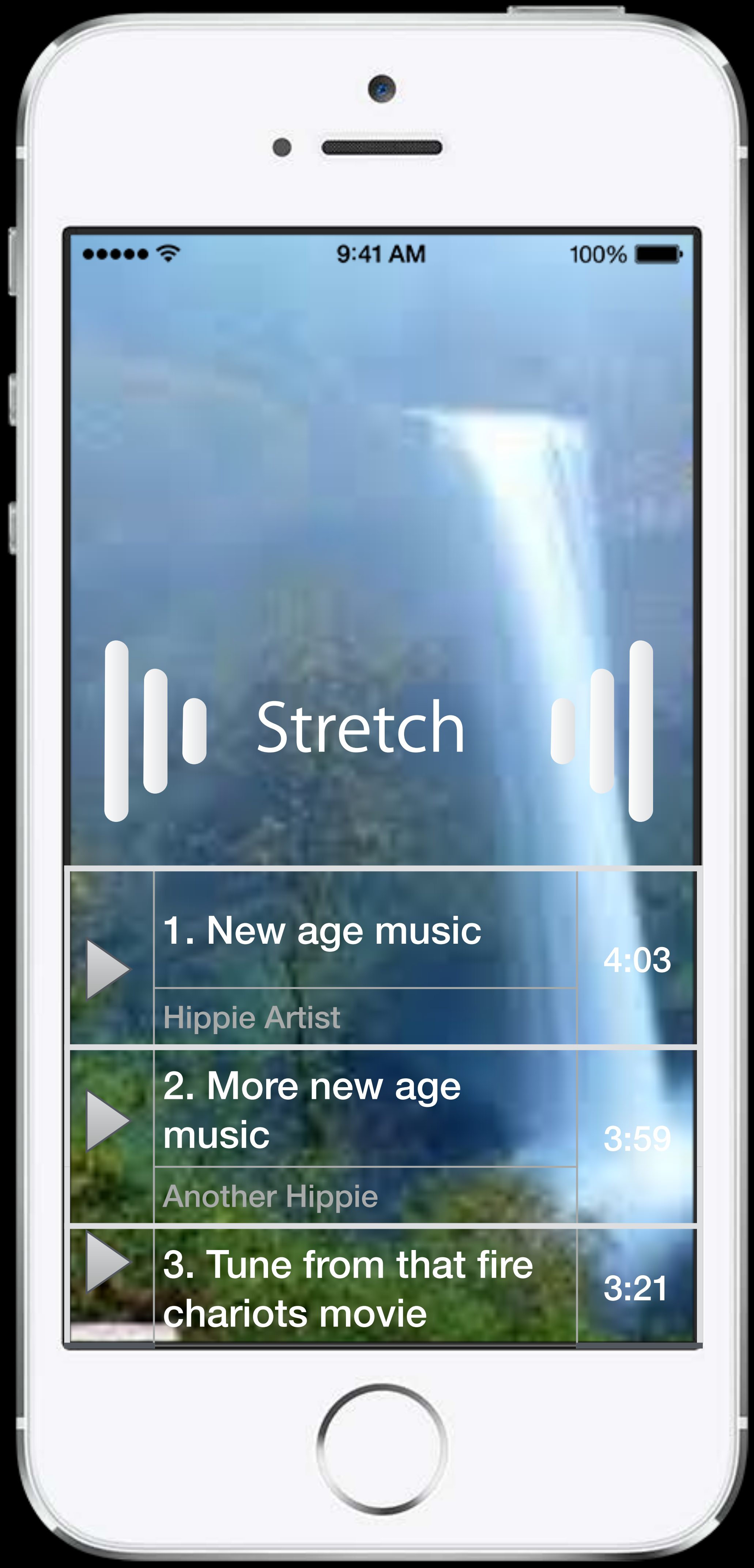
- Automatically change my experience throughout workout
- Provide real-time feedback to me as I run
- Summarize my performance when I complete

Running App

Warm up

Running App

Warm up



9:41 AM

100%

Stretch

1. New age music

Hippie Artist

4:03

2. More new age music

Another Hippie

3:59

3. Tune from that fire chariots movie

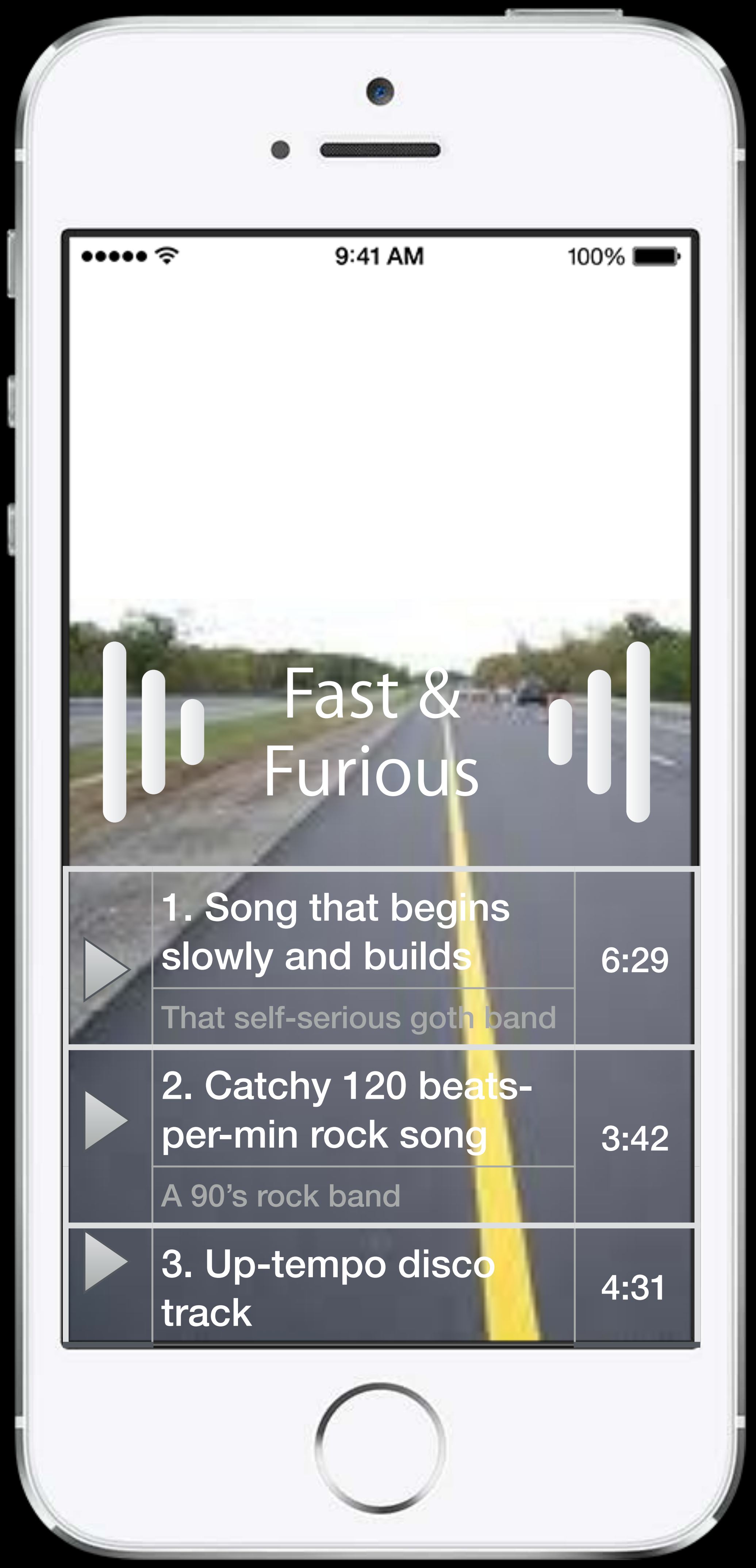
3:21

Running App

Exercise

Running App

Exercise



9:41 AM

100%

Fast & Furious

1. Song that begins slowly and builds
That self-serious goth band
6:29

2. Catchy 120 beats-per-min rock song
A 90's rock band
3:42

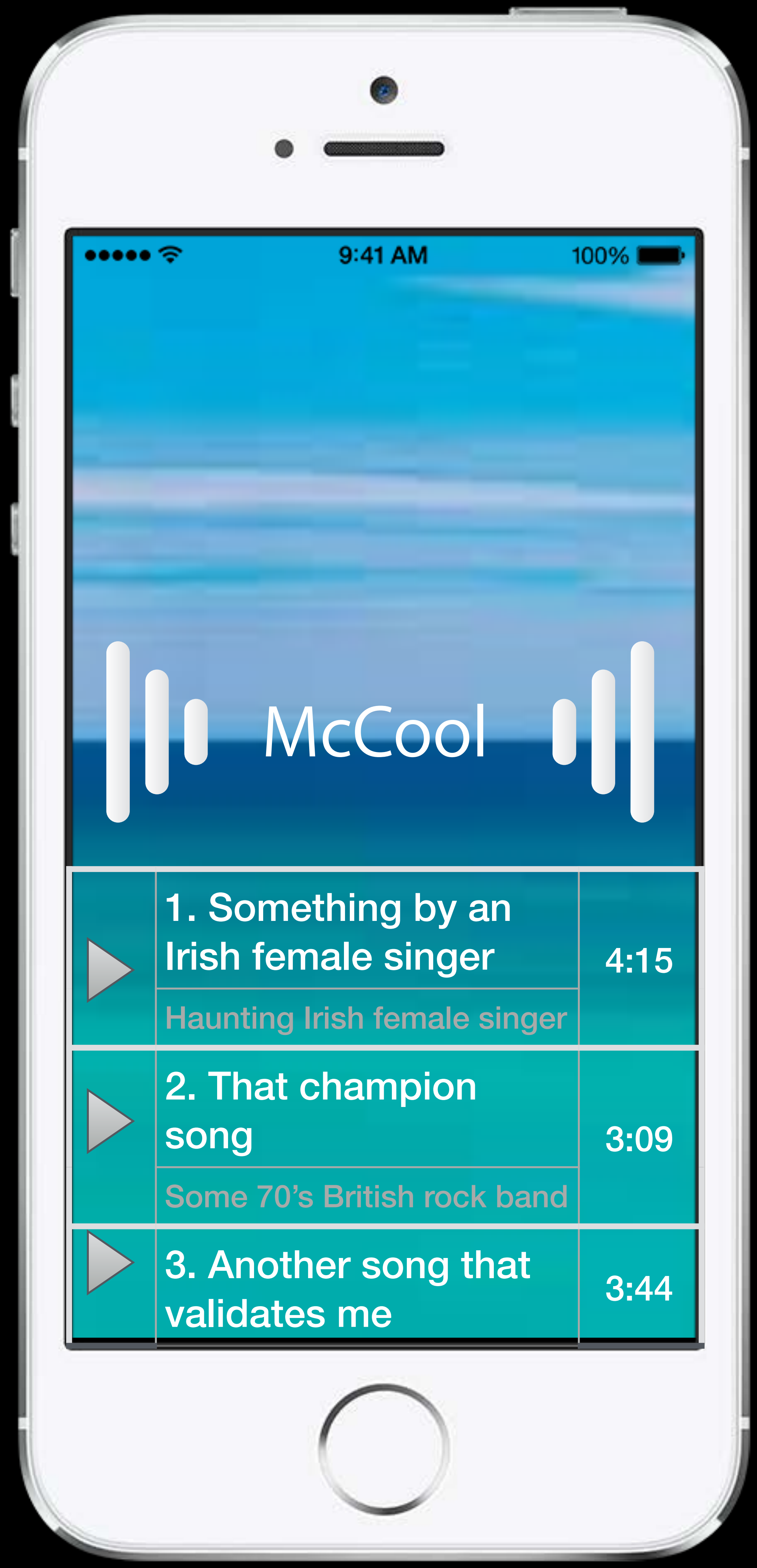
3. Up-tempo disco track
4:31

Running App

Cool-down

Running App

Cool-down



9:41 AM 100%

McCool

▶	1. Something by an Irish female singer Haunting Irish female singer	4:15
▶	2. That champion song Some 70's British rock band	3:09
▶	3. Another song that validates me	3:44

Running App

Cool-down

Pull Interface

Journaling app example

Pull Interface

Journaling app example

What I would want in a journaling app

Pull Interface

Journaling app example

What I would want in a journaling app

- Do most of my work for me

Pull Interface

Journaling app example

What I would want in a journaling app

- Do most of my work for me
- Correlate my motion activity with other sources of information

Pull Interface

Journaling app example

What I would want in a journaling app

- Do most of my work for me
- Correlate my motion activity with other sources of information
- Profile my physical activity

Journaling App

Journaling App

Get the location context

Journaling App

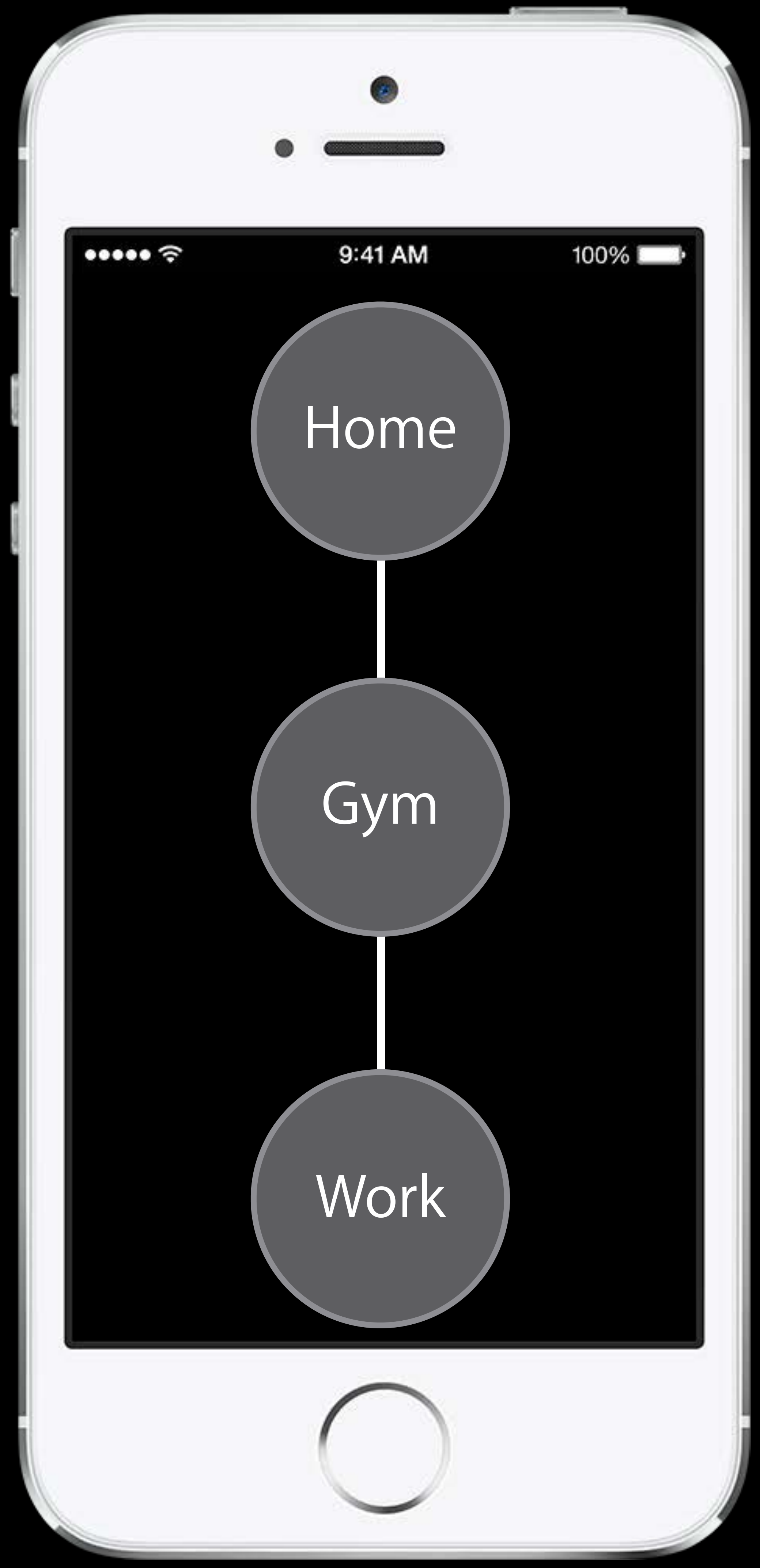
Get the location context

- Query Visit Monitoring to get list of places visited throughout day

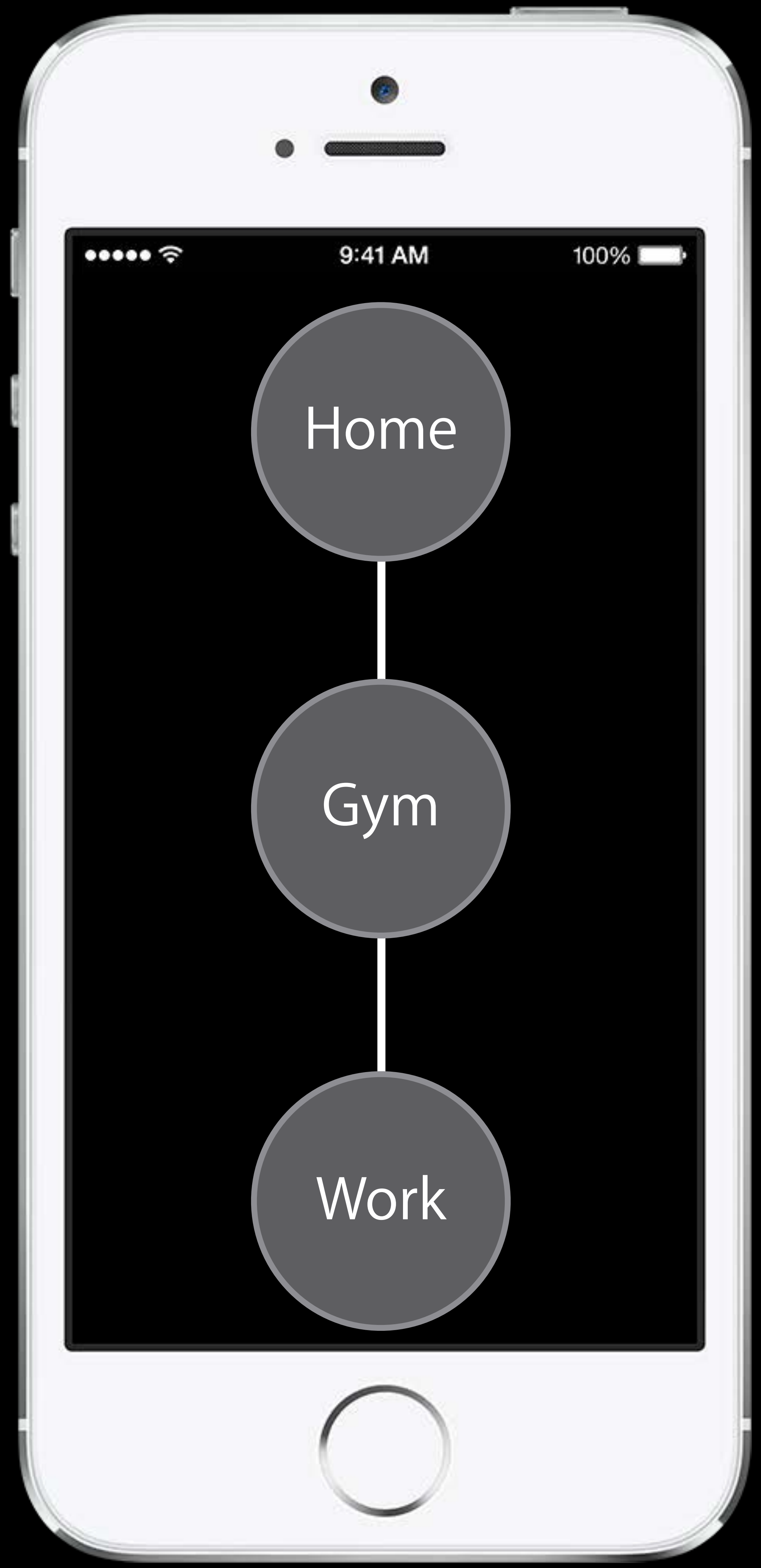
Journaling App

Get the location context

- Query Visit Monitoring to get list of places visited throughout day

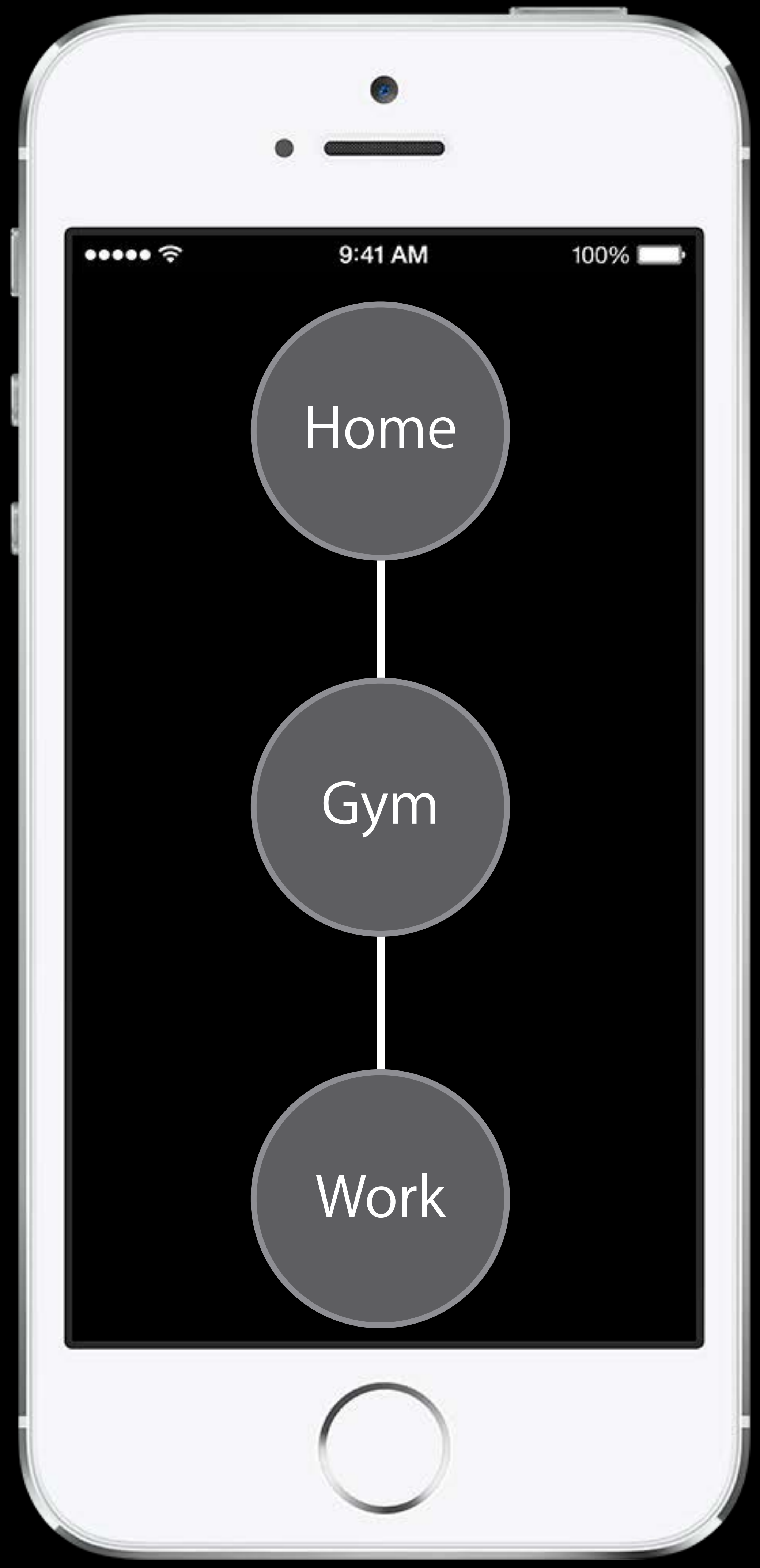


Journaling App



Journaling App

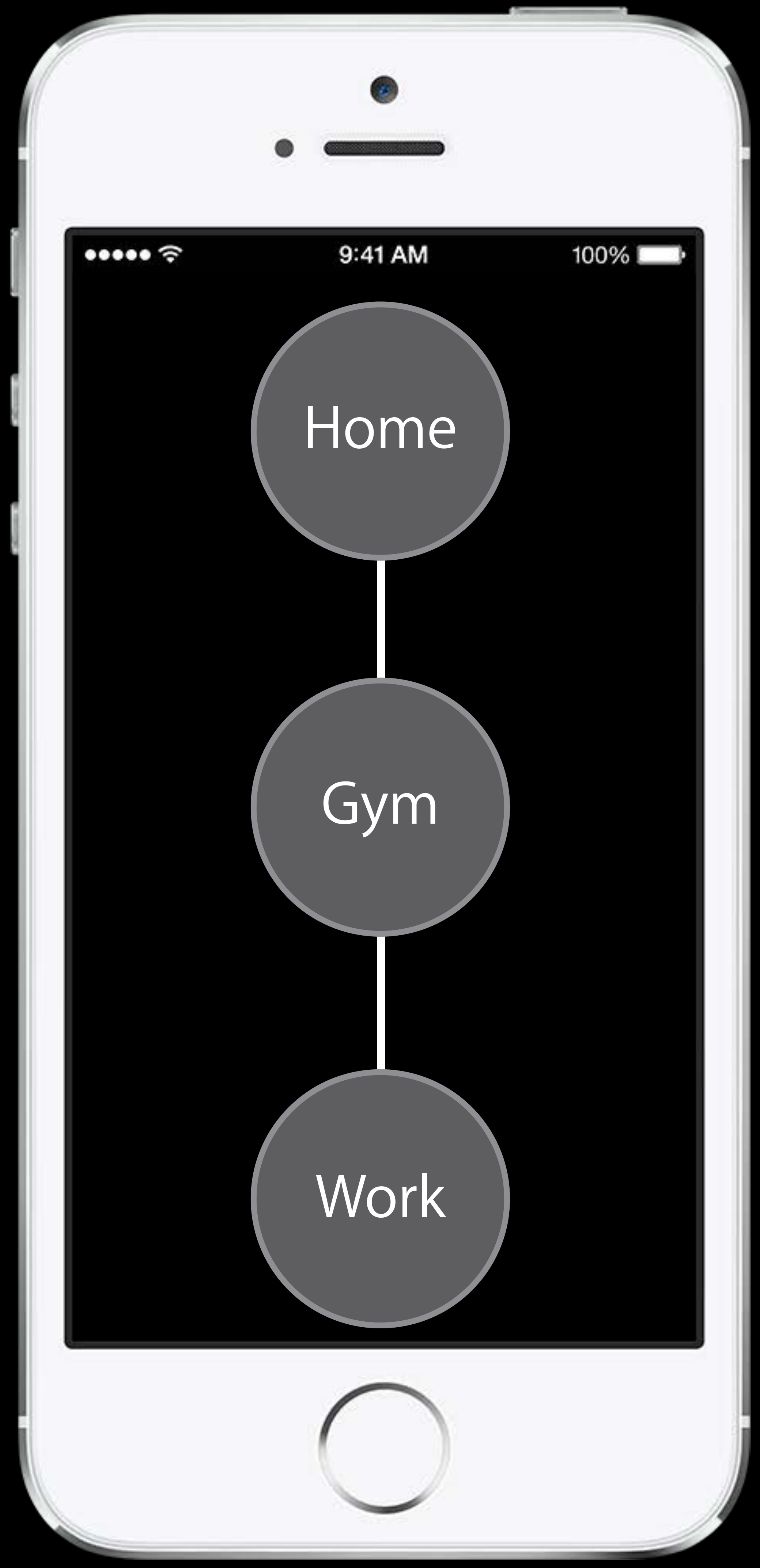
Get the transport context



Journaling App

Get the transport context

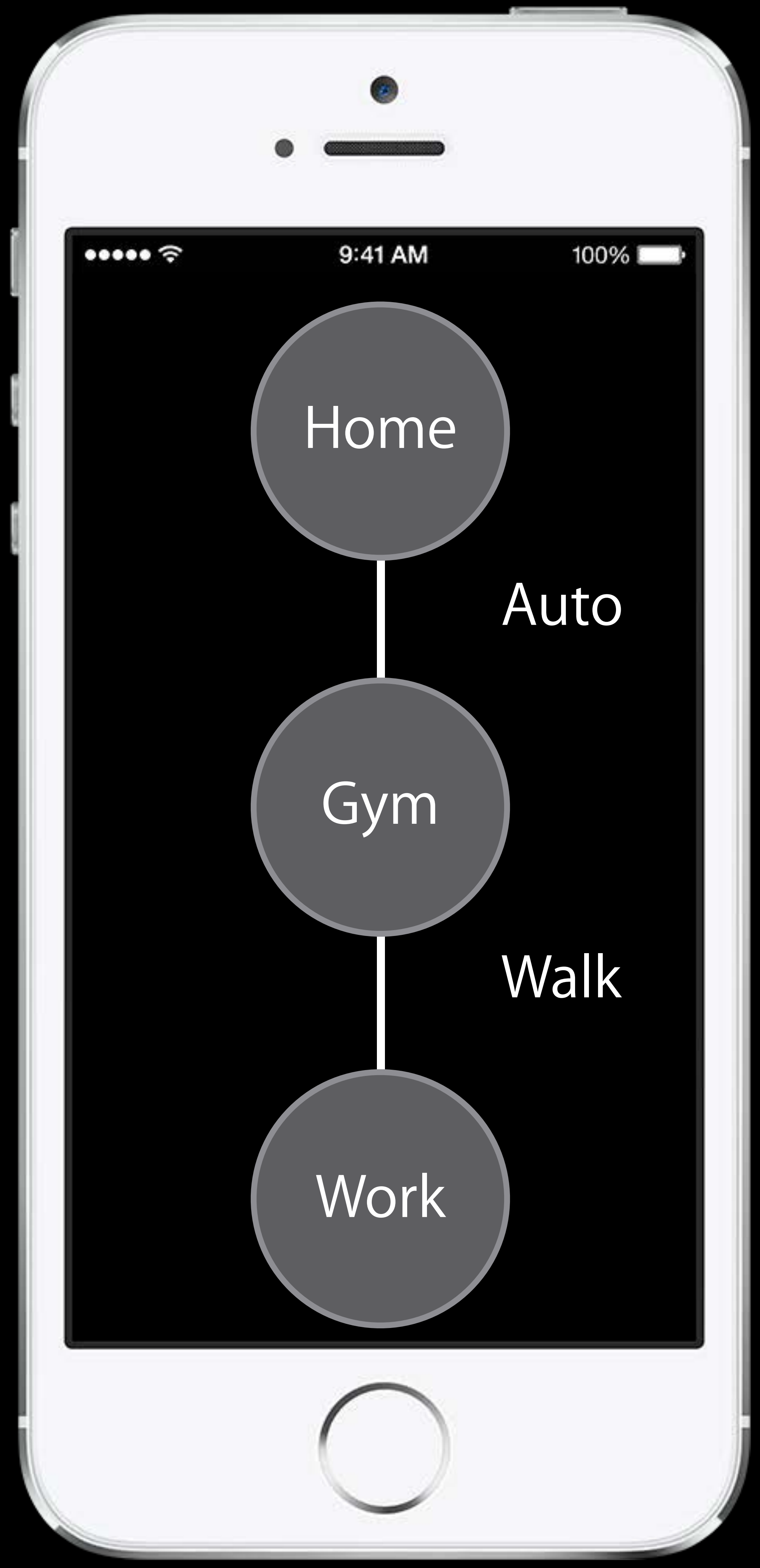
- Query Motion Activity to obtain the travel mode between locations



Journaling App

Get the transport context

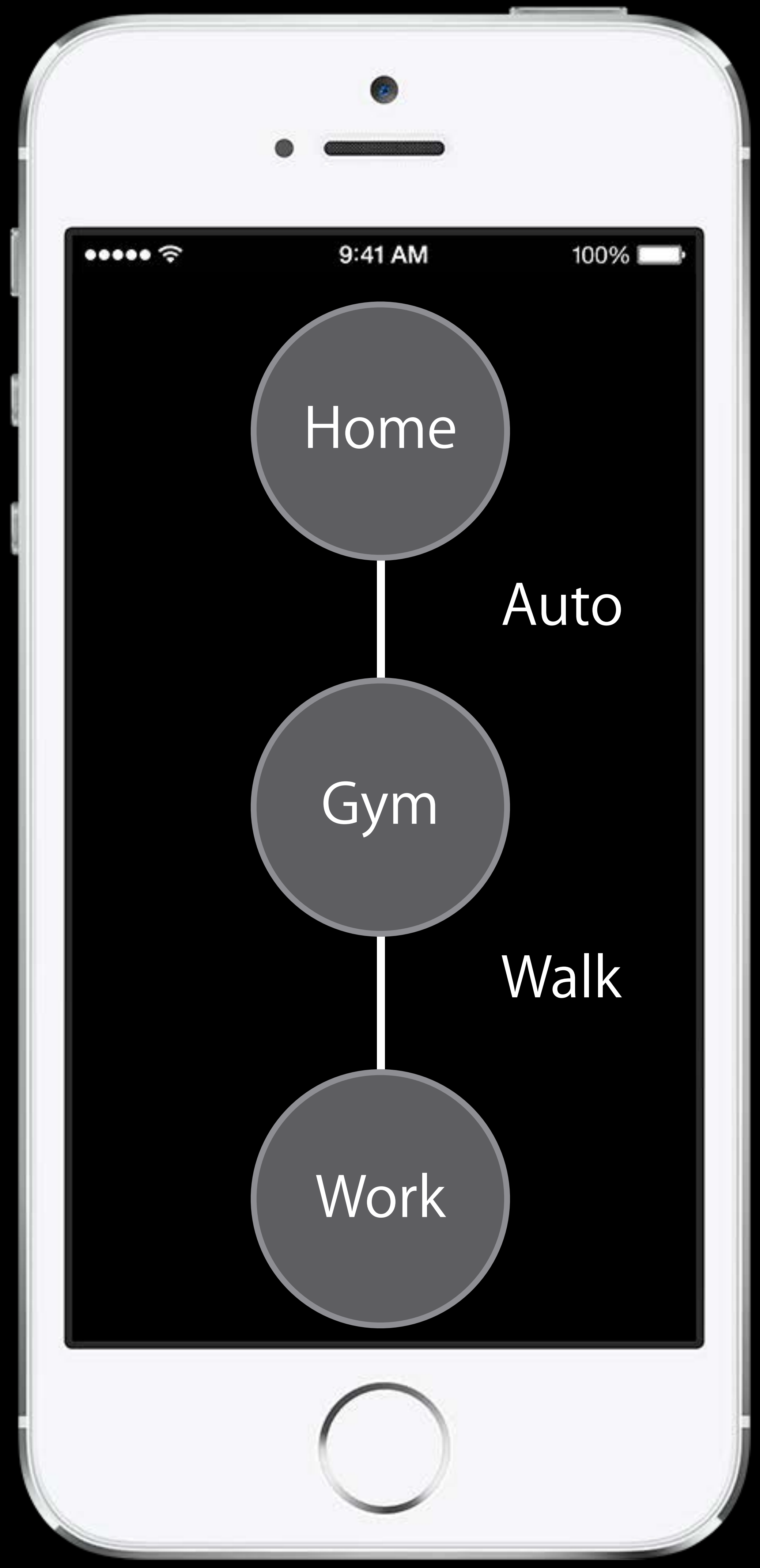
- Query Motion Activity to obtain the travel mode between locations



Journaling App

Get the transport context

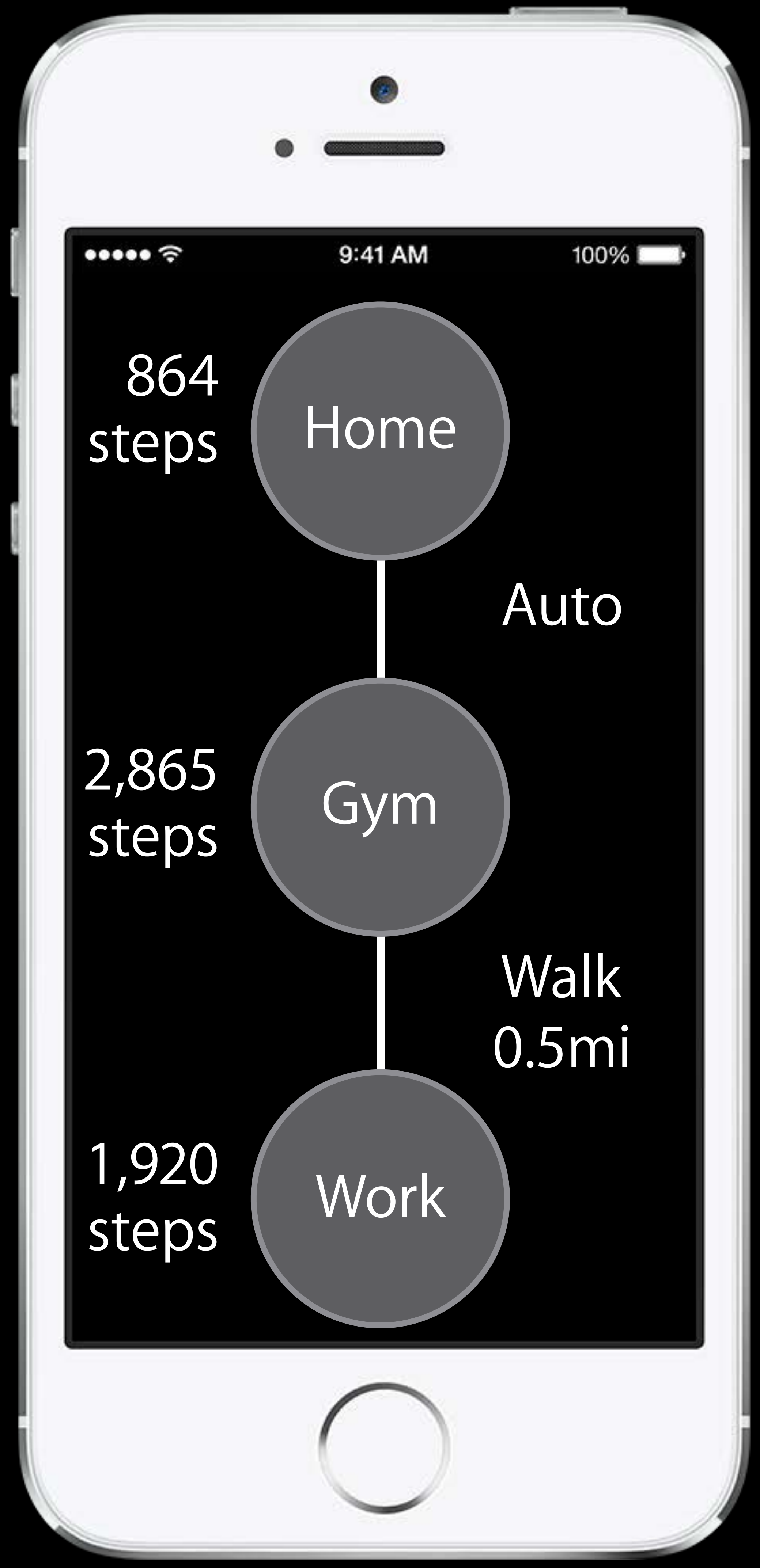
- Query Motion Activity to obtain the travel mode between locations
- Where appropriate, query Pedometer to get
  - Distance traveled between two locations
  - Steps taken at each location



Journaling App

Get the transport context

- Query Motion Activity to obtain the travel mode between locations
- Where appropriate, query Pedometer to get
  - Distance traveled between two locations
  - Steps taken at each location



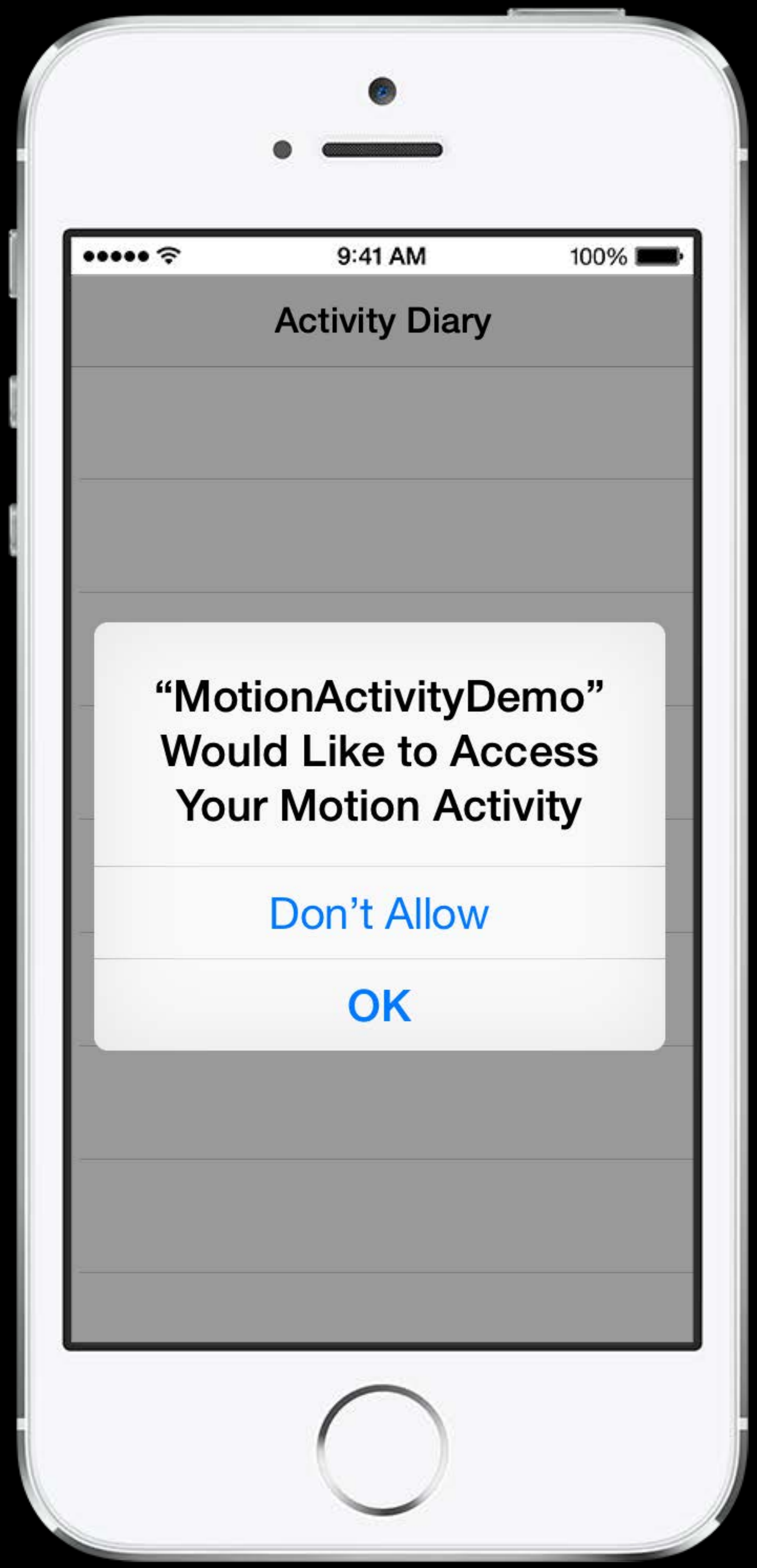
Deep Dive

Privacy

User needs to opt in

Privacy

User needs to opt in

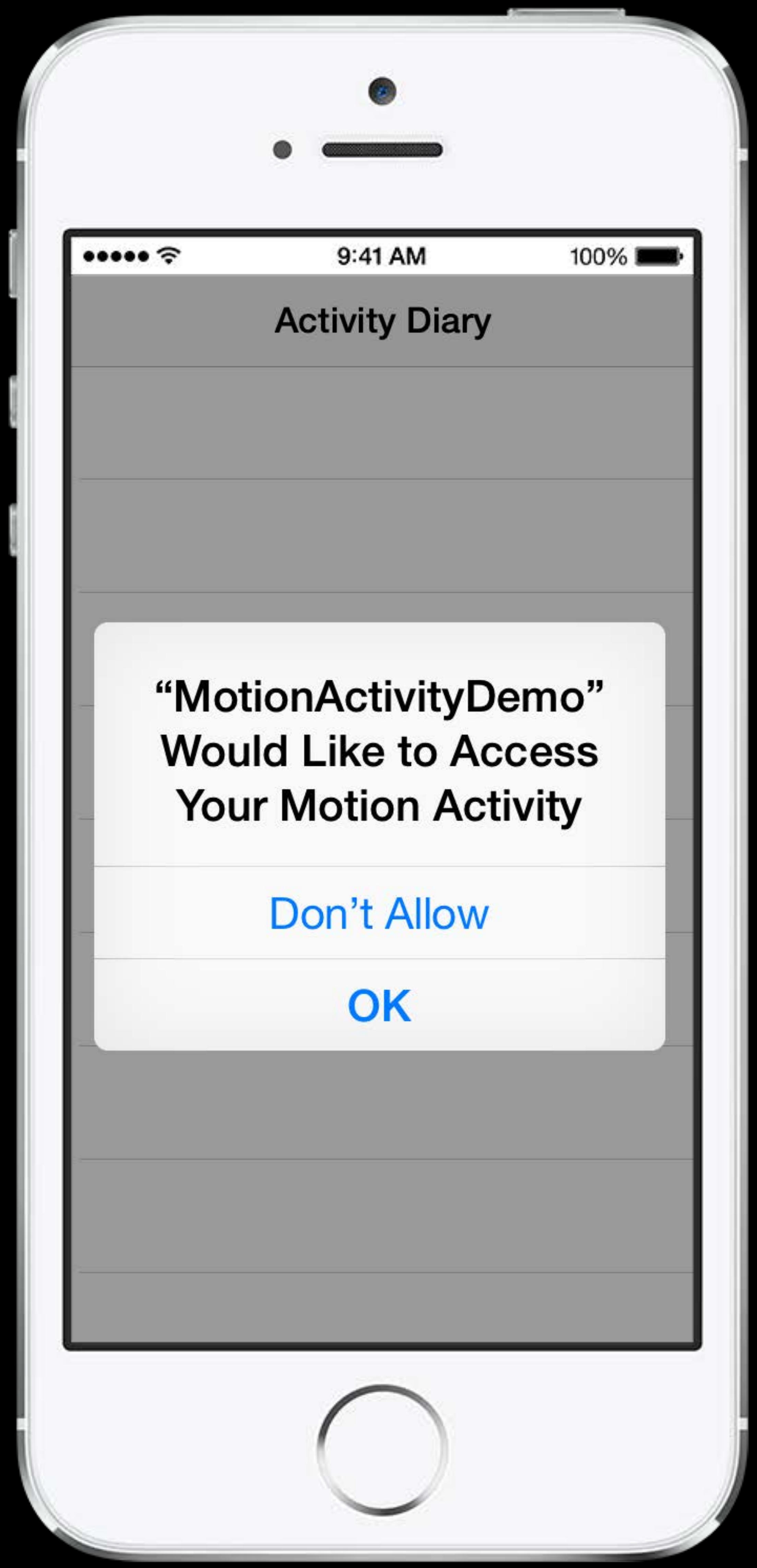


“MotionActivityDemo” Would Like to Access Your Motion Activity

Privacy

User needs to opt in

- Data availability from seven days prior



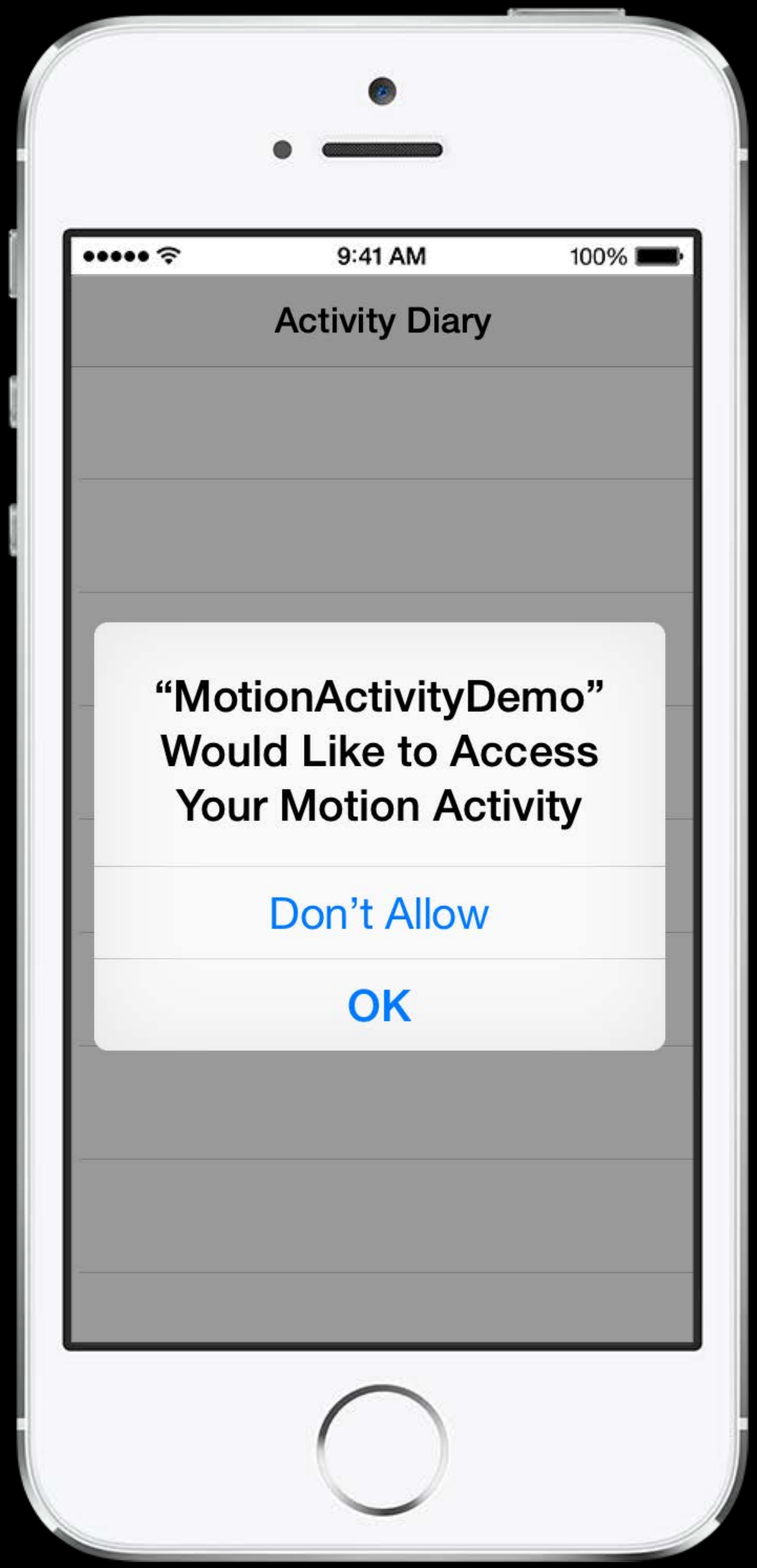
"MotionActivityDemo" Would Like to Access Your Motion Activity

Privacy

User needs to opt in

- Data availability from seven days prior

User can opt out at any time



“MotionActivityDemo” Would Like to Access Your Motion Activity

Activity Classifier API

Checking for availability

`NSObject CMMotionActivityManager`

- `+ (BOOL)isActivityAvailable`

Practice good hygiene

Activity Classifier API

Checking for availability

```
NSObject CMMotionActivityManager
• + (BOOL)isActivityAvailable
```

Practice good hygiene

Activity Classifier API

Push interface

```
NSObject CMMotionActivityManager
• – (void)startActivityUpdatesToQueue:(NSOperationQueue *)queue
                            withHandler:(CMMotionActivityHandler)handler
• – (void)stopActivityUpdates
```

1. An update with current activity will arrive first
2. Handler will be called on subsequent activity changes

Activity Classifier API

Push interface

```
NSObject CMMotionActivityManager
• – (void)startActivityUpdatesToQueue:(NSOperationQueue *)queue
                              withHandler:(CMMotionActivityHandler)handler
• – (void)stopActivityUpdates
```

1. An update with current activity will arrive first
2. Handler will be called on subsequent activity changes

Activity Classifier API

Pull interface

```
NSObject CMMotionActivityManager
• – (void)queryActivityStartingFromDate:(NSDate *)start
                                 toDate:(NSDate *)end
                                toQueue:(NSOperationQueue *)queue
                            withHandler:(CMMotionActivityQueryHandler)
```

Queries for activity transitions that have occurred during the specified time range. Results are returned to the handler on the provided queue.

1. First activity returned may have a start-date before the specified start.
2. This activity represents what the state was at the start time.

Activity Classifier API

Pull interface

`NSObject CMMotionActivityManager`

```
• – (void)queryActivityStartingFromDate:(NSDate *)start
                                 toDate:(NSDate *)end
                                toQueue:(NSOperationQueue *)queue
                            withHandler:(CMMotionActivityQueryHandler)
```

Queries for activity transitions that have occurred during the specified time range. Results are returned to the handler on the provided queue.

1. First activity returned may have a start-date before the specified start.
2. This activity represents what the state was at the start time.

Activity Classifier API

Activity transition start

`NSObject CMMotionActivity`

- `NSDate *startDate`

The time at which the activity started

Activity Classifier API

Activity transition start

```
NSObject CMMotionActivity
• NSDate *startDate
```

The time at which the activity started

Activity Classifier API

Confidence level

```
NSObject CMMotionActivity
• NSDate *startDate
• CMMotionActivityConfidence confidence

enum CMMotionACtivityConfidence {
    CMMotionActivityConfidenceLow,
    CMMotionActivityConfidenceMedium,
    CMMotionActivityConfidenceHigh }
```

Confidence allows you to trade off accuracy for responsiveness

Activity Classifier API

Confidence level

```
NSObject CMMotionActivity
• NSDate *startDate
• CMMotionActivityConfidence confidence
```

```
enum CMMotionACtivityConfidence {
    CMMotionActivityConfidenceLow,
    CMMotionActivityConfidenceMedium,
    CMMotionActivityConfidenceHigh }
```

Confidence allows you to trade off accuracy for responsiveness

Activity Classifier API

Stationary device

`NSObject CMMotionActivity`

- `BOOL stationary`

TRUE if device is **not** moving

Activity Classifier API

Stationary device

```
NSObject CMMotionActivity
• BOOL stationary
```

TRUE if device is **not** moving

Activity Classifier API

Activity types

`NSObject CMMotionActivity`

- `BOOL stationary`
- `BOOL walking`
- `BOOL running`
- `BOOL automotive`
- `BOOL cycling`
- `BOOL unknown`

Mutually exclusive motion activity types

Activity Classifier API

Activity types

NSObject CMMotionActivity

- BOOL stationary
- BOOL walking
- BOOL running
- BOOL automotive
- BOOL cycling
- BOOL unknown

Mutually exclusive motion activity types

Example Scenarios

Example Scenarios

Device scenarios	`stationary`	`walking`	`running`	`automotive`	`cycling`	`unknown`

Example Scenarios

Device scenarios	stationary	walking	running	automotive	cycling	unknown
On table	true	false	false	false	false	false

Example Scenarios

Device scenarios	stationary	walking	running	automotive	cycling	unknown
On table	true	false	false	false	false	false
On runner’s upper arm	false	false	true	false	false	false

Example Scenarios

Device scenarios	stationary	walking	running	automotive	cycling	unknown
On table	true	false	false	false	false	false
On runner's upper arm	false	false	true	false	false	false
In dash of idling vehicle	true	false	false	true	false	false

Example Scenarios

Device scenarios	stationary	walking	running	automotive	cycling	unknown
On table	true	false	false	false	false	false
On runner’s upper arm	false	false	true	false	false	false
In dash of idling vehicle	true	false	false	true	false	false
In dash of moving vehicle	false	false	false	true	false	false

Example Scenarios

Device scenarios	stationary	walking	running	automotive	cycling	unknown
On table	true	false	false	false	false	false
On runner's upper arm	false	false	true	false	false	false
In dash of idling vehicle	true	false	false	true	false	false
In dash of moving vehicle	false	false	false	true	false	false
Passenger checking email	false	false	false	false	false	false

Example Scenarios

Device scenarios	stationary	walking	running	automotive	cycling	unknown
On table	true	false	false	false	false	false
On runner's upper arm	false	false	true	false	false	false
In dash of idling vehicle	true	false	false	true	false	false
In dash of moving vehicle	false	false	false	true	false	false
Passenger checking email	false	false	false	false	false	false
Immediately after reboot	false	false	false	false	false	true

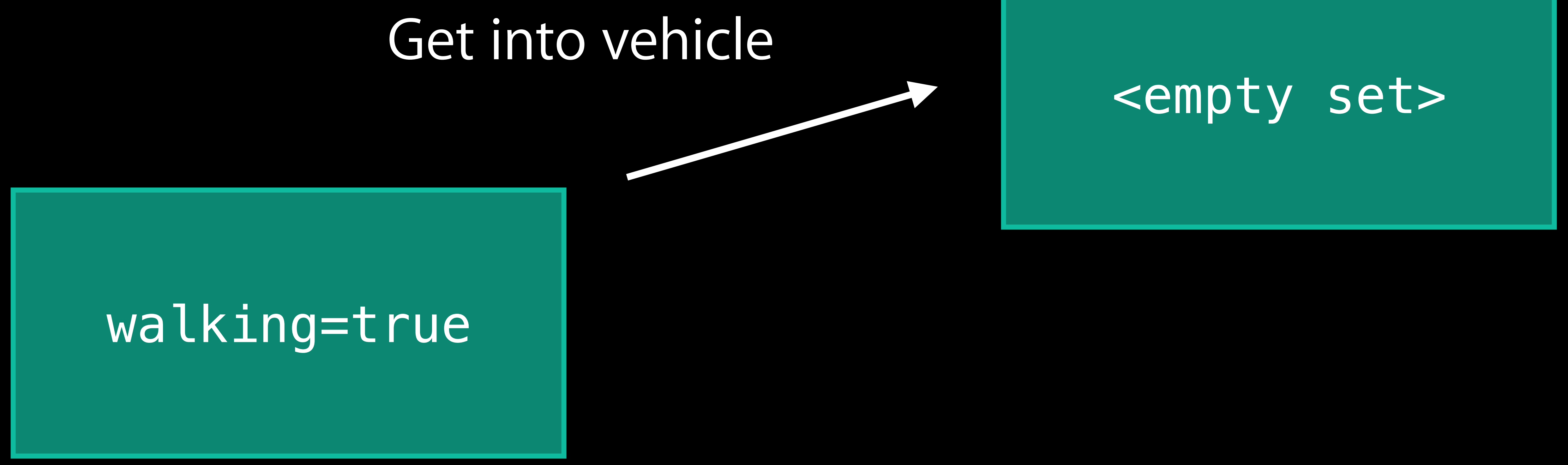
Example Scenarios

Device scenarios	stationary	walking	running	automotive	cycling	unknown
On table	true	false	false	false	false	false
On runner's upper arm	false	false	true	false	false	false
In dash of idling vehicle	true	false	false	true	false	false
In dash of moving vehicle	false	false	false	true	false	false
Passenger checking email	false	false	false	false	false	false
Immediately after reboot	false	false	false	false	false	true
In zumba class	false	false	false	false	false	false

Driving Example

walking=true

Driving Example

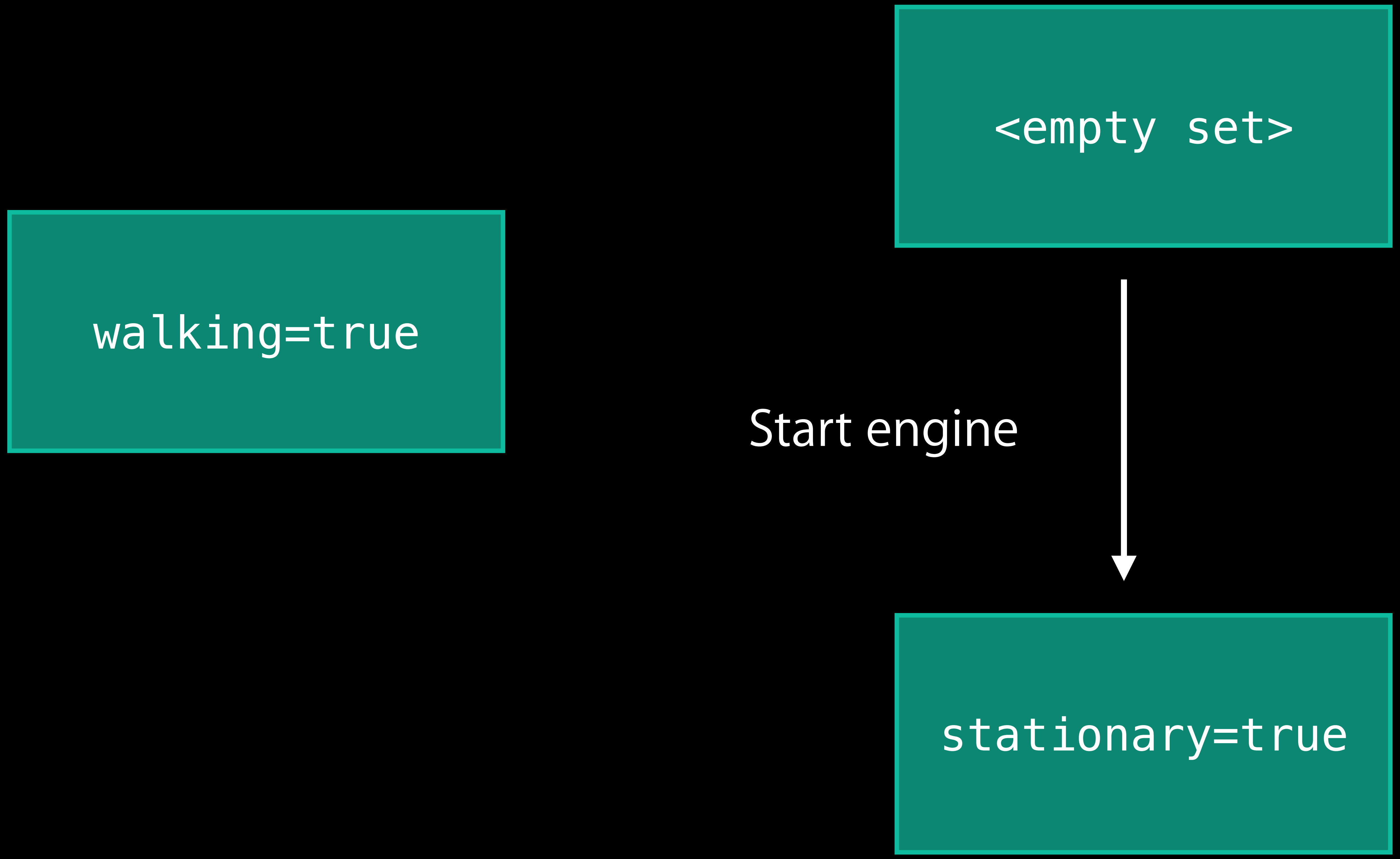


Get into vehicle

`<empty set>`

`walking=true`

Driving Example



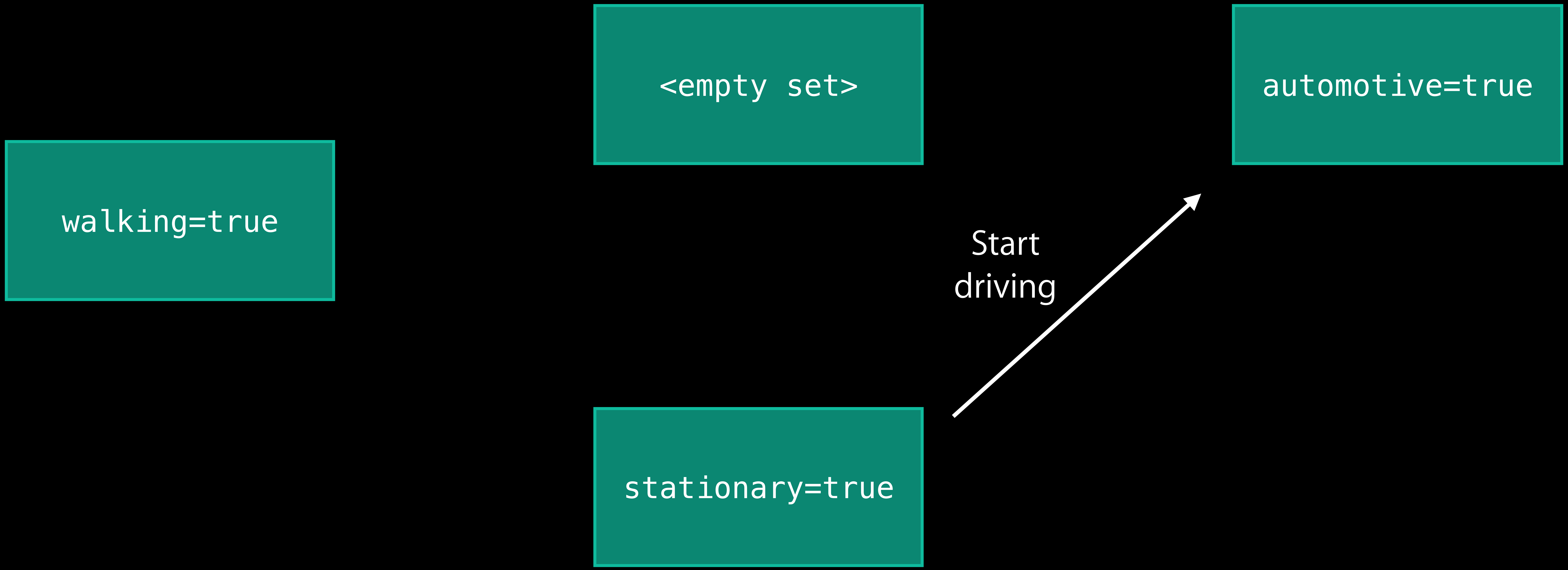
walking=true

<empty set>

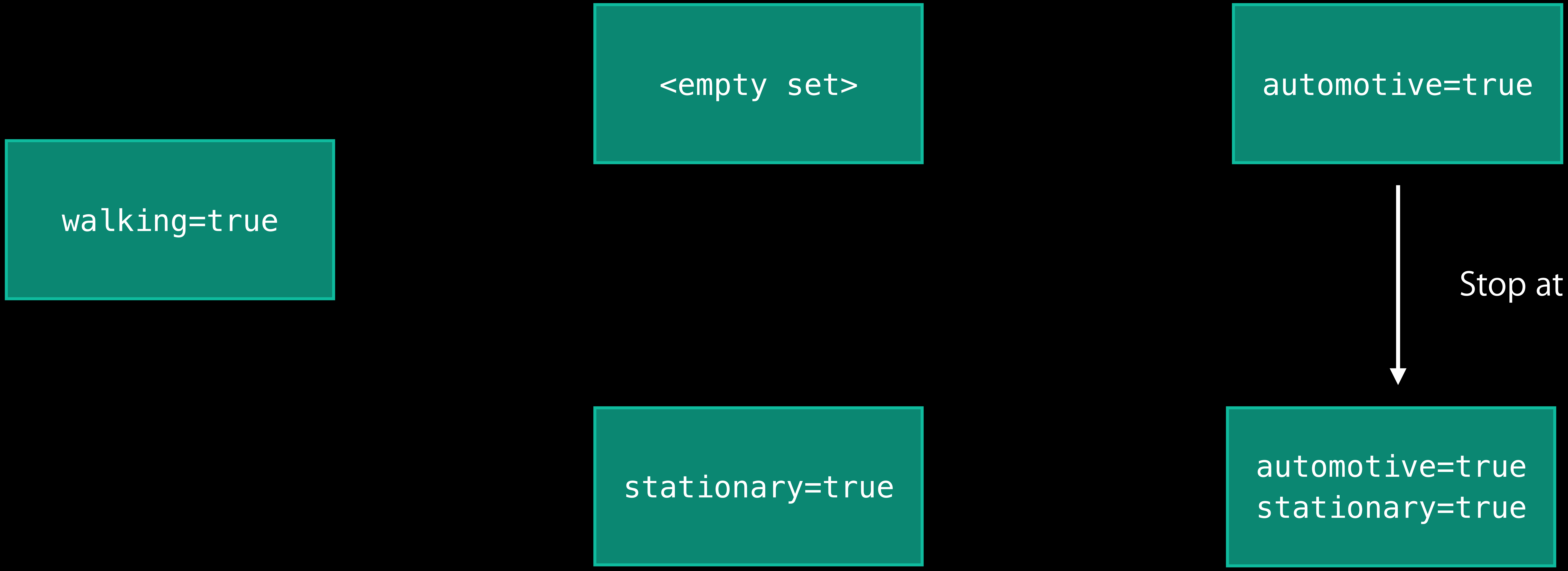
Start engine

stationary=true

Driving Example



Driving Example



walking=true

<empty set>

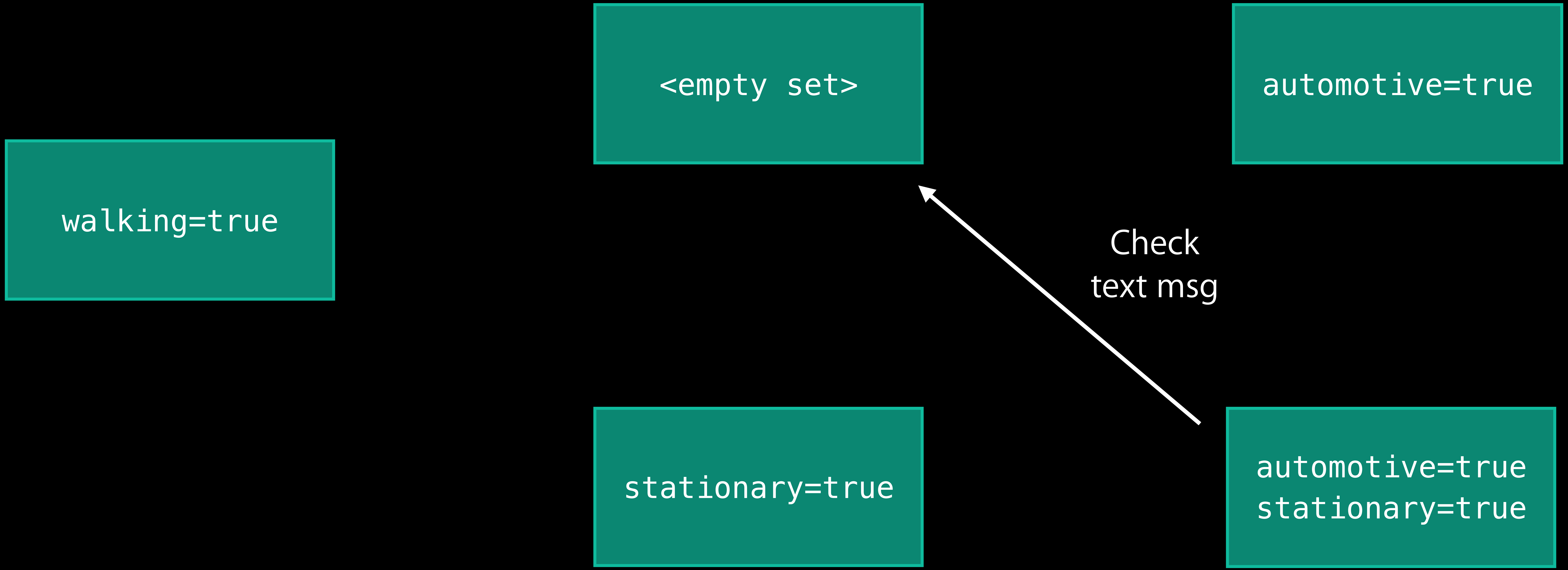
stationary=true

automotive=true

Stop at light

automotive=true
stationary=true

Driving Example



walking=true

<empty set>

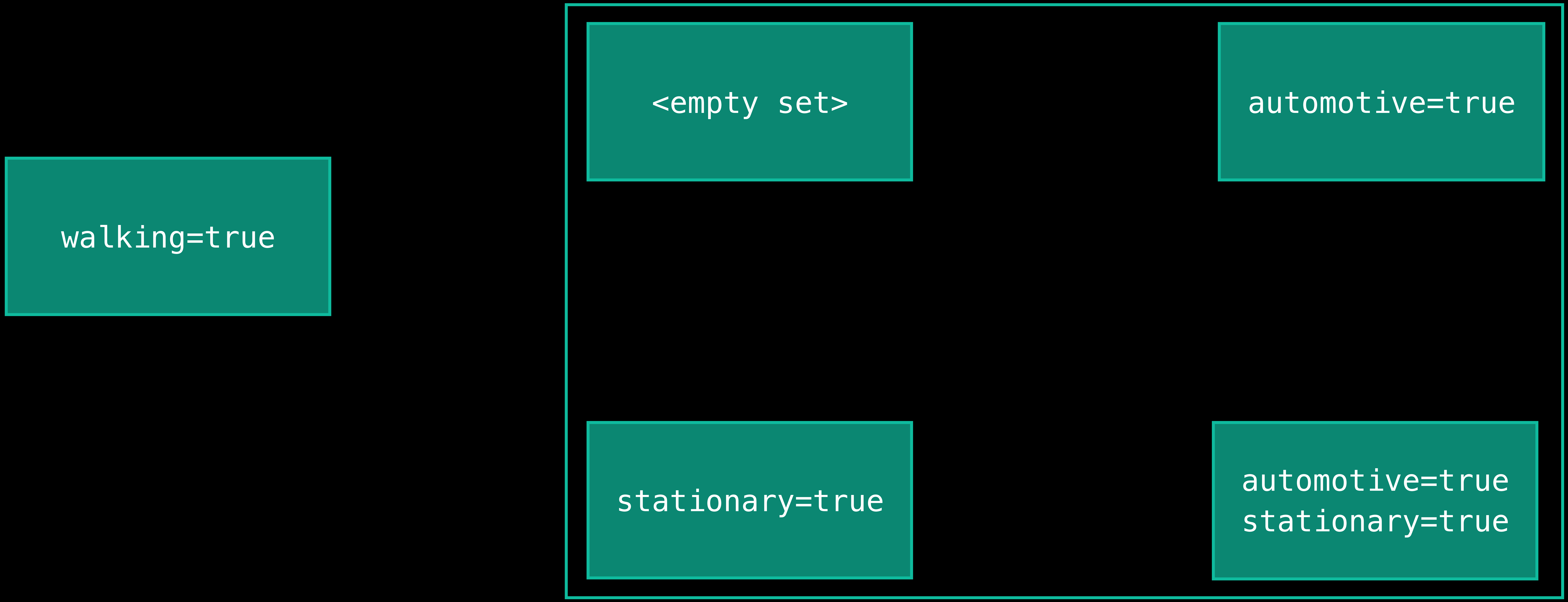
automotive=true

Check text msg

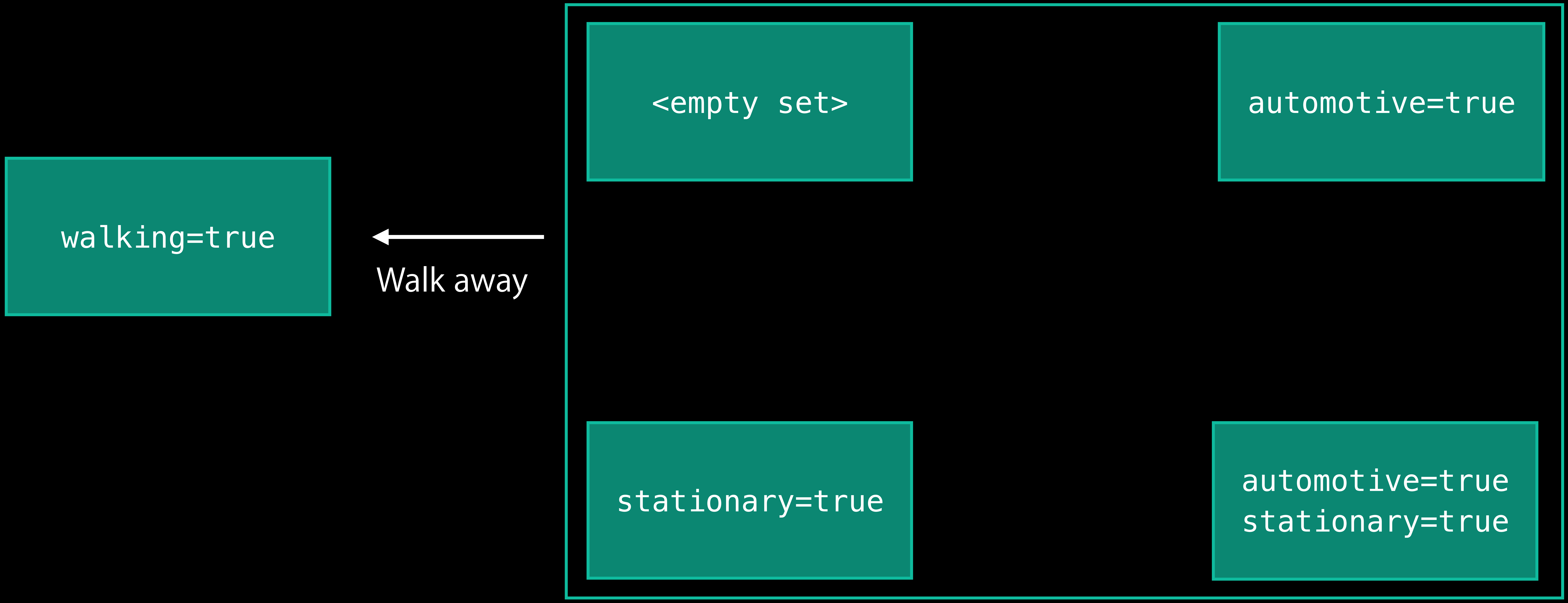
stationary=true

automotive=true
stationary=true

Driving Example



Driving Example



Pedometer API

Checking for step-counting availability

`NSObject CMPedometer`

- `+ (BOOL)isStepCountingAvailable`
- `+ (BOOL)isDistanceAvailable`

Pedometer API

Checking for step-counting availability

```
NSObject CMPedometer
• + (BOOL)isStepCountingAvailable
• + (BOOL)isDistanceAvailable
```

Pedometer API

Push interface

```
NSObject CMPedometer
• -(void)startPedometerUpdatesFromDate:(NSDate *)start
                           withHandler:(CMPedometerHandler)handler
• -(void)stopPedometerUpdates
```

Starts a series of regular pedometer updates to the handler on a serial queue

1. Each update contains cumulative pedestrian activity since the specified start date
2. Update is sent every 2.5s if steps have changed
3. After app is resumed from the background, it will immediately receive an update

Pedometer API

Push interface

NSObject CMPedometer

```
• –(void)startPedometerUpdatesFromDate:(NSDate *)start
                                withHandler:(CMPedometerHandler)handler
• –(void)stopPedometerUpdates
```

Starts a series of regular pedometer updates to the handler on a serial queue

1. Each update contains cumulative pedestrian activity since the specified start date
2. Update is sent every 2.5s if steps have changed
3. After app is resumed from the background, it will immediately receive an update

Pedometer API

Pull interface

```
NSObject CMPedometer
• -(void)queryPedometerDataFromDate:(NSDate *)start
                             toDate:(NSDate *)end
                        withHandler:(CMPedometerHandler)handler
```

Queries for user’s pedestrian activity in specified time range

1. App will receive a single record containing entire pedometer activity between start and end dates

Pedometer API

Pull interface

NSObject CMPedometer

```
• -(void)queryPedometerDataFromDate:(NSDate *)start
                             toDate:(NSDate *)end
                        withHandler:(CMPedometerHandler)handler
```

Queries for user's pedestrian activity in specified time range

1. App will receive a single record containing entire pedometer activity between start and end dates

Pedometer API

Pedometer record timestamp

`NSObject CMPedometerData`

- `NSDate *startDate`
- `NSDate *endDate`

Pedometer API

Pedometer record timestamp

NSObject CMPedometerData

- NSDate *startDate
- NSDate *endDate

Pedometer API

Cumulative number of steps

`NSObject CMPedometerData`

- `NSDate *startDate`
- `NSDate *endDate`
- `NSNumber *numberOfSteps`

Number of steps taken between startDate and endDate

Pedometer API

Cumulative number of steps

NSObject CMPedometerData

- `NSDate *startDate`
- `NSDate *endDate`
- `NSNumber *numberOfSteps`

Number of steps taken between startDate and endDate

Pedometer API

Cumulative distance

`NSObject CMPedometerData`

- `NSDate *startDate`
- `NSDate *endDate`
- `NSNumber *numberOfSteps`
- `NSNumber *distance`

Total distance (in meters) traveled by the user between startDate and endDate

Pedometer API

Cumulative distance

```
NSObject CMPedometerData
• NSDate *startDate
• NSDate *endDate
• NSNumber *numberOfSteps
• NSNumber *distance
```

Total distance (in meters) traveled by the user between startDate and endDate

Demo

Let’s code!

Sunny Chow

Engineer

Demo Topics

`CMMotionActivityManager (iOS 7.0+)`
`CMPedometer (iOS 8.0+)`

Coding sample is available now!

More Information

Allan Schaffer
Graphics and Game Technologies Evangelist
aschaffer@apple.com

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

Related Sessions

● Introducing HealthKit	Mission	Tuesday 10:15AM
● What’s New in Core Location	Marina	Tuesday 2:00PM

Labs

Core Motion Lab	Graphics and Games Lab A	Friday 12:45PM

WWDC14