

What's New in Core Location

Session 500

Morgan Grainger

Software Engineer

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

Why Are You Here?

You want to...

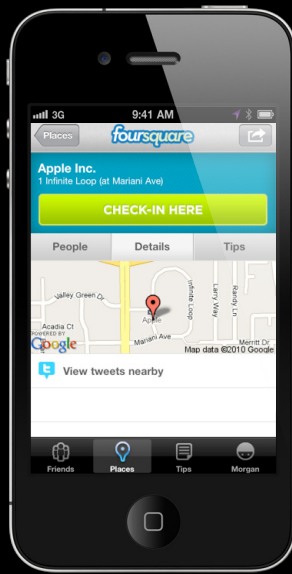
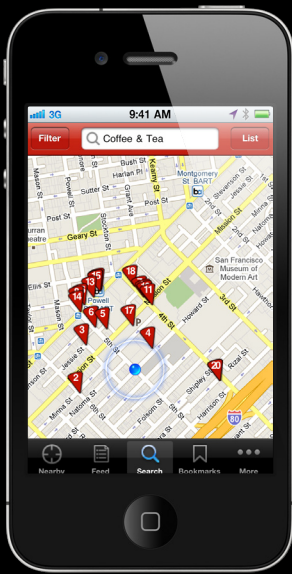
- Make your applications easier to use, more intuitive
- Get started on a new location-aware app

Why Is Location Powerful?

Location provides context

- iOS: inherently mobile
- More intuitive applications
- Enhances the user experience
- Enables new use cases





Agenda

- Using the Core Location services
- Authorization: putting the user in control
- What's new in Core Location?
 - Making location data more meaningful

Using Core Location

Choose the right service for your use case

Three Positioning Methods



Cell



Wi-Fi

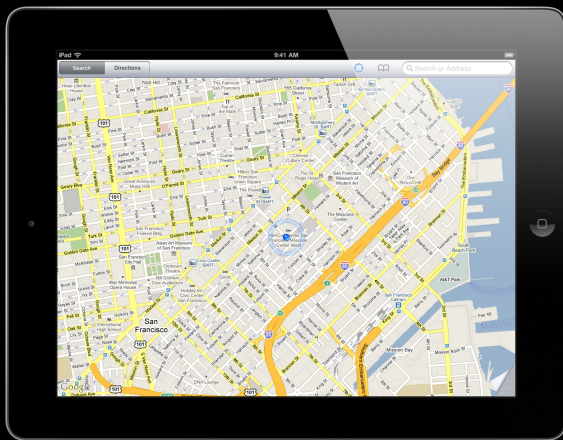


GPS

Core Location Services

- Standard location
- Significant change
- Region monitoring

Standard Location Service



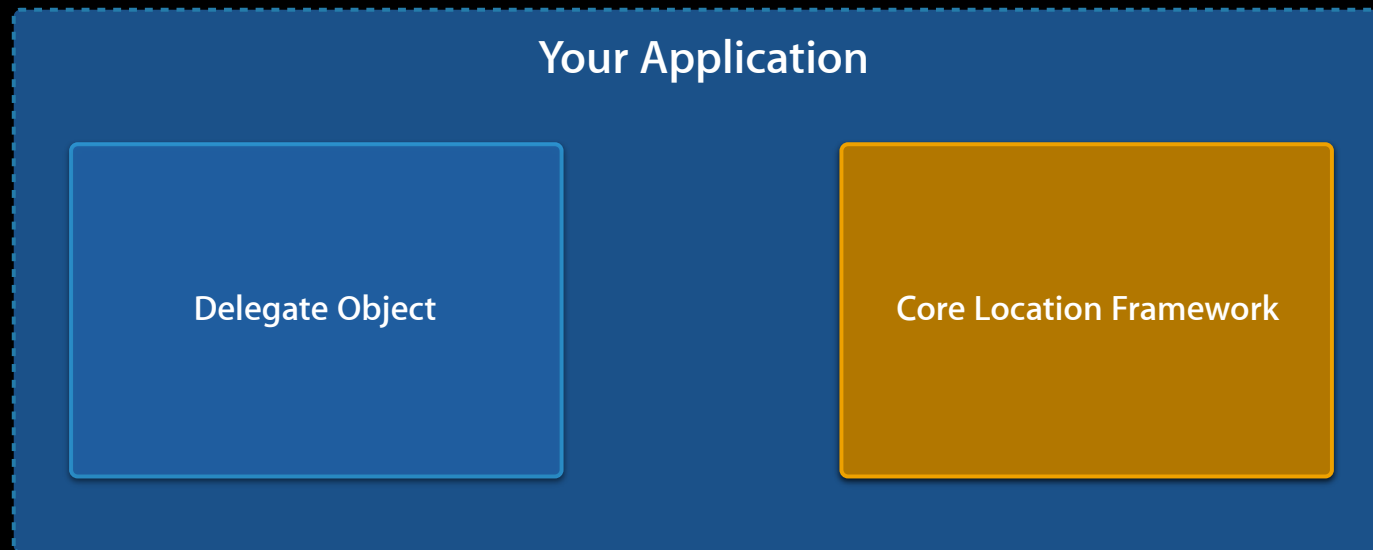
Current Location



Navigation

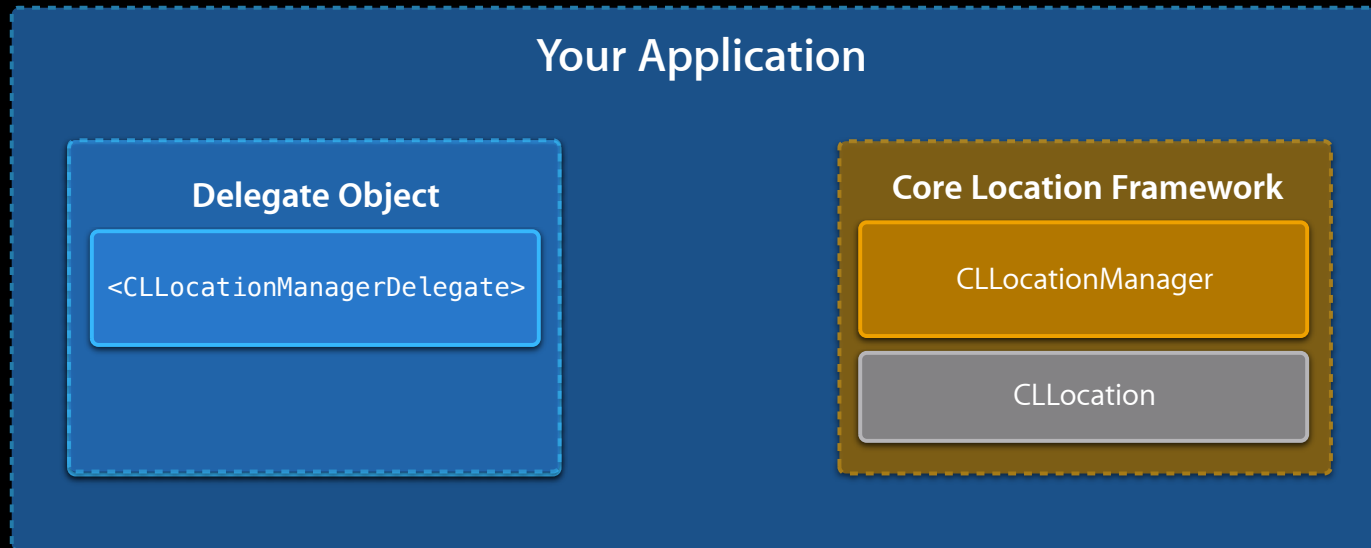
Core Location Architecture

Primary components



Core Location Architecture

Primary components



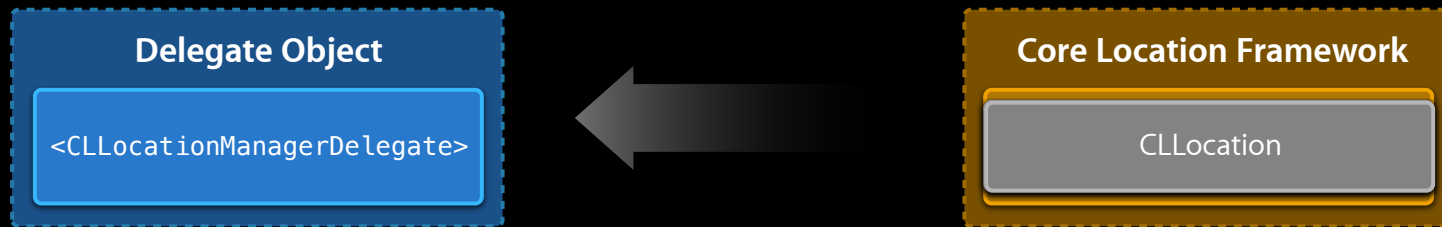
Using Core Location

Starting location updates



Using Core Location

Receiving location updates

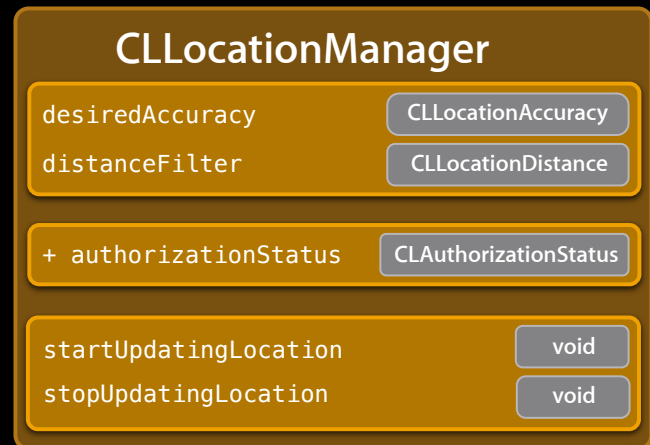


```
locationManager:didUpdateToLocation:fromLocation:
```

Using Core Location

Configuring the location manager

- Specify the accuracy you require
- Distance filter prevents unneeded callbacks
- Check if location services are enabled

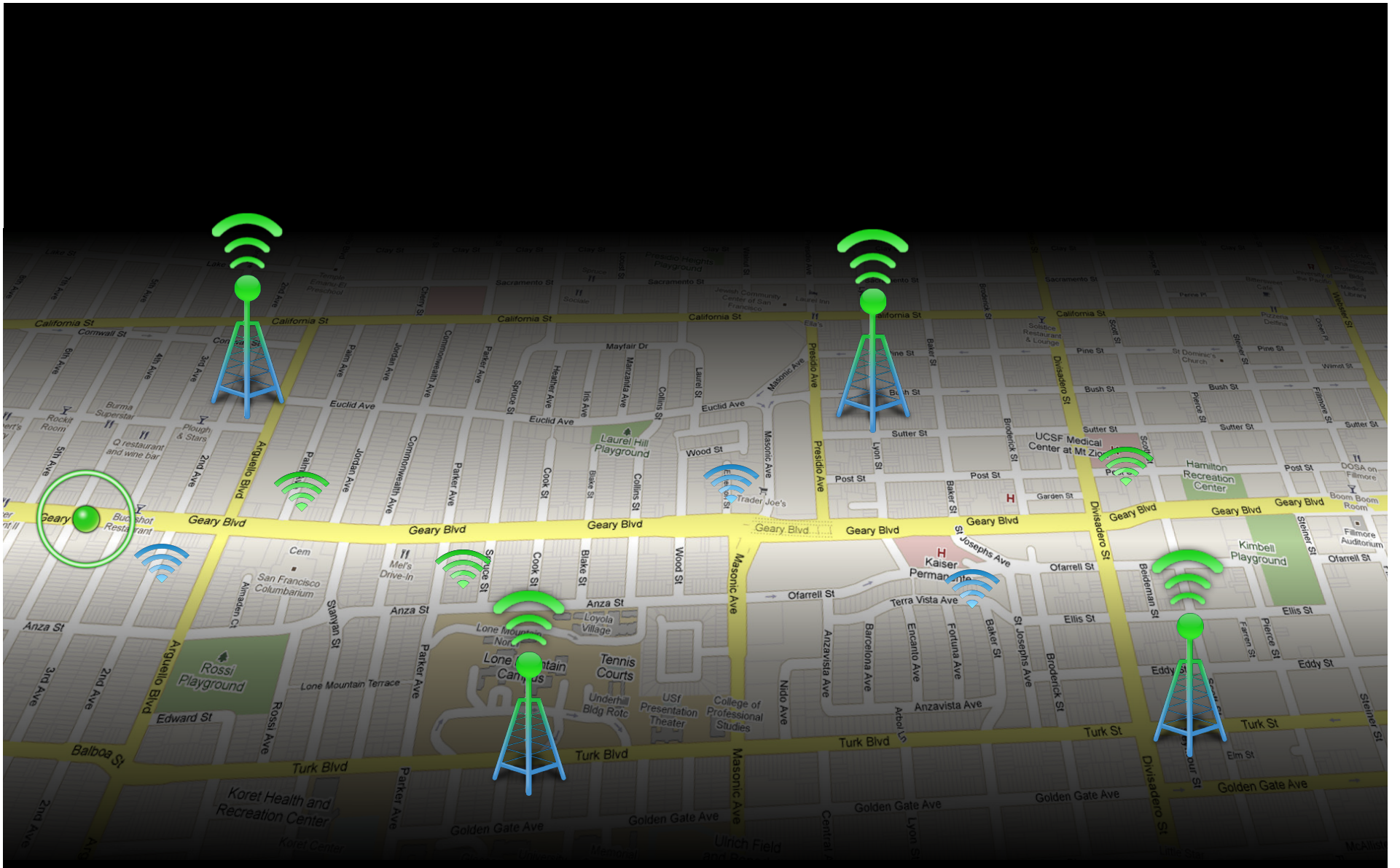


Continuous Location Applications

- User expects the same experience whether or not the application is frontmost
- Declare in Info.plist

```
<key>UIBackgroundModes</key>  
<array>  
    <string>location</string>  
</array>
```





Significant Location Change

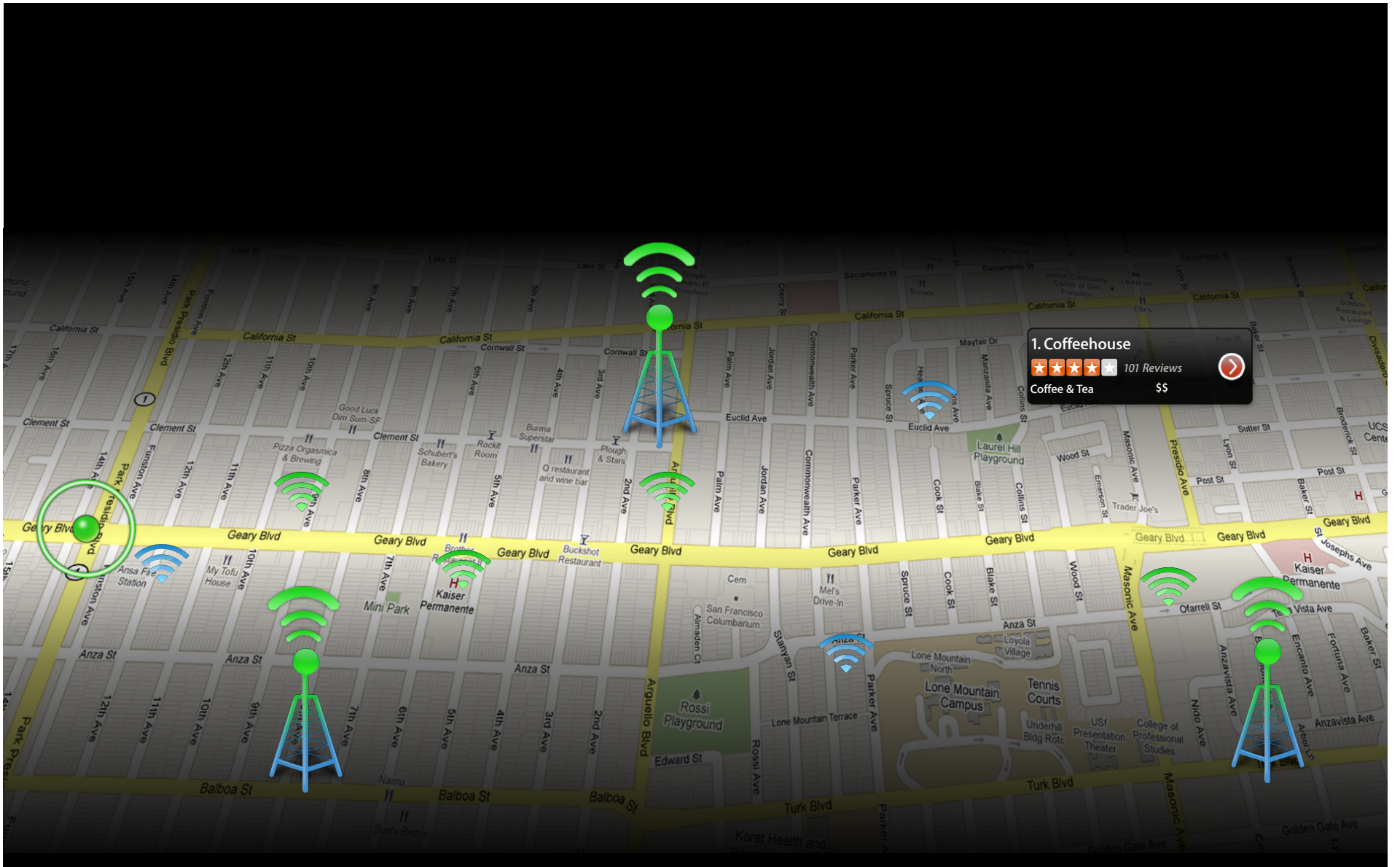
- Calculates location when device moves significantly
- Application will be launched in the background if not running
- iOS 5: improved accuracy

Significant Location Change Monitoring



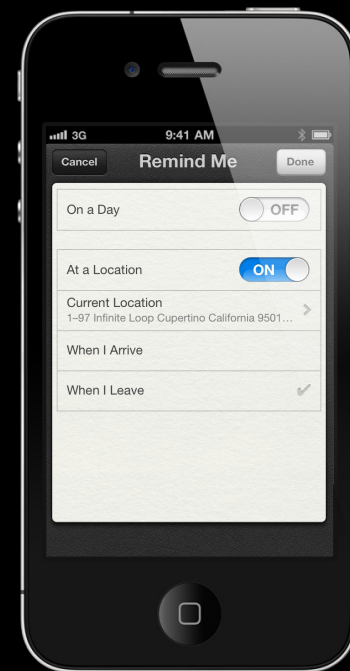
```
startMonitoringSignificantLocationChanges
```

```
locationManager:didUpdateToLocation:fromLocation:
```



Region Monitoring API

- Register regions surrounding locations of interest
- Notified when user enters or exits region
- Application will be launched in the background if not running
- Based on cell positioning



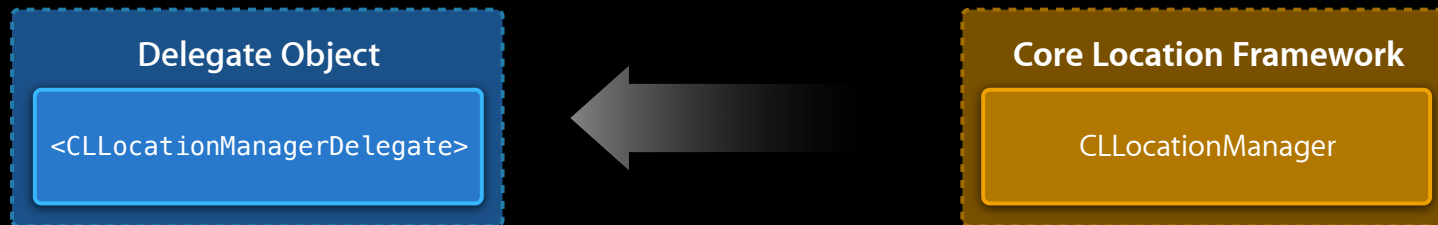
Starting Region Monitoring

```
CLLocationCoordinate2D coord = CLLocationCoordinate2DMake(37.332426, -122.030404);  
CLRegion *region = [[CLRegion alloc] initWithCenter:coord  
                  radius:1000.0  
                  identifier:@"Apple Inc."];
```



```
startMonitoringForRegion:(CLRegion *)region;
```


Receiving Region Monitoring Events

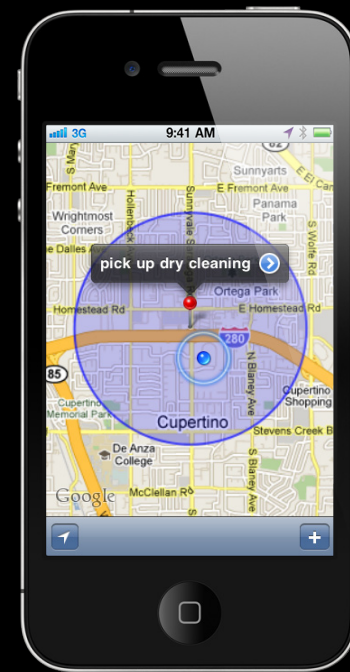


`locationManager:didEnterRegion:`

`locationManager:didExitRegion:`

Improvements to Region Monitoring

- Uses system events as triggers
- Makes use of every type of position, if available



Putting the User in Control

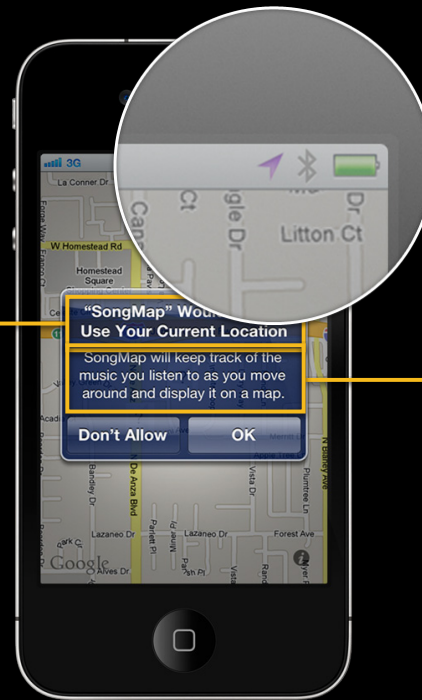
Straight Talk About Location Authorization

- Not all users will enable location services for your app
- Use the API to provide context
- Help the user understand the consequences of their decision

Authorization

Help the user make an informed decision

Permission Dialog
Shown once per application; plan for "Don't Allow"



Status bar icon

Visible when an application is using the user's location

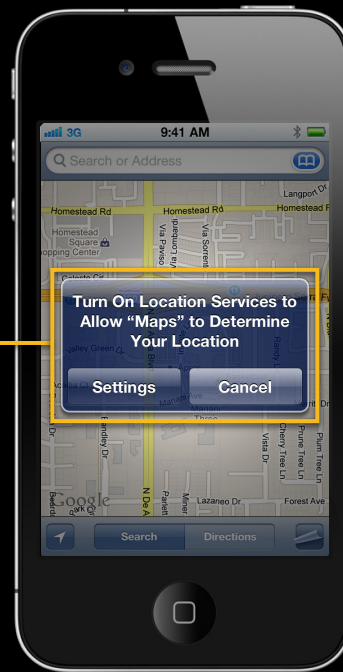
Purpose property

Tell the user why your application needs to use location services

Authorization

Help the user make an informed decision

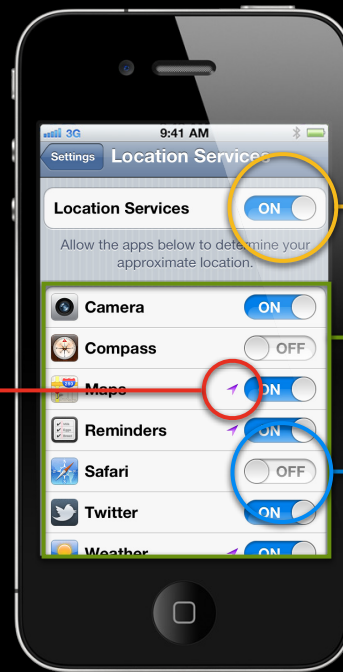
Re-Enable Dialog
Presented by the
system; will not
always be shown



Authorization

The user is in control

Location arrow
Displayed if the application has requested location in the last 24 hours



Enabled switch
Retrieve value from location manager class function

List of applications
Applications appear after requesting location services

Approval switch
Check approval status using CLLocationManager

Checking Authorization States

Put the user in control

- Respond appropriately to the user's location preferences

Authorization State	Description
<code>kCLAuthorizationStatusNotDetermined</code>	User not yet prompted
<code>kCLAuthorizationStatusAuthorized</code>	Your application is authorized to use location services
<code>kCLAuthorizationStatusDenied</code>	The user has denied the use of location services for your application or for all applications
<code>kCLAuthorizationStatusRestricted</code>	The user cannot authorize your application

What's New?

Making location data more meaningful

Coordinates Are Only Part of the Story

Core Location Framework

CLLocation

37.78338, -122.403354

Reverse Geocoding

Transforms a coordinate into an address

800 Howard St.
San Francisco, CA



37.78338, -122.403354

Forward Geocoding

Transforms an address into a coordinate

800 Howard St.
San Francisco, CA



37.78338, -122.403354

Use Cases for Geocoding

- Describe the current location
- Locate an address book contact
- Tag notes with the city where they were written



CLGeocoder

- New class in Core Location
- Forward and reverse geocoding
- Fully asynchronous
- One CLGeocoder per request
 - Create more if necessary

CLPlacemark

- All geocoding requests return a CLPlacemark
 - Normalized location
 - Region
 - Address dictionary
 - Descriptive properties
- Integrates with the Address Book framework

Using CLGeocoder

- Define a completion handler to handle the result of the geocode

```
typedef void (^CLGeocodeCompletionHandler)(NSArray *placemarks, NSError *error);
```

- Array of placemarks returned on success
- NSError returned on failure

Why Might a Geocode Fail?

- Network error
- No results
- Partial result only
- Request canceled

CLPlacemark and the Address Book

They work together

- Turn a contact's address into a location
- Store a returned address in the address book
- Format a returned address appropriately for the device's current locale

Geocoding Demo

Richard Heard
iOS SDK Engineer

Reverse Geocoding

```
CLGeocoder *geocoder = [[CLGeocoder alloc] init];
CLLocation *location = [[CLLocation alloc] initWithCoordinate:
                        CLLocationCoordinate2DMake(37.33,-122.03)];
```

```
[geocoder reverseGeocodeLocation:location completionHandler:
                        ^(NSArray *placemarks, NSError *error) {
    for (CLPlacemark *placemark in placemarks) {
        NSLog(@"Placemark %@", placemark);
    }
    [geocoder release];
}]
```

```
[location release];
```

Forward Geocoding

Using an address string

```
CLGeocoder *geocoder = [[CLGeocoder alloc] init];
```

```
NSString *addressString = @"1 Infinite Loop, Cupertino";
```

```
[geocoder geocodeAddressString:addressString completionHandler:  
    ^(NSArray *placemarks, NSError *error) {  
    for (CLPlacemark *placemark in placemarks) {  
        NSLog(@"Placemark %@", placemark);  
    }  
    [geocoder release];  
}];
```

```
[location release];
```

Forward Geocoding

Using an address dictionary

```
#import <AddressBook/ABCGlobals.h>

CLGeocoder *geocoder = [[CLGeocoder alloc] init];
NSDictionary *address = [NSDictionary dictionaryWithObjectsAndKeys:
    @"1 Infinite Loop", kABCAddressStreetKey,
    @"Cupertino", kABCAddressCityKey,
    @"California", kABCAddressStateKey, nil];
[geocoder geocodeAddressDictionary:address completionHandler:
    ^(NSArray *placemarks, NSError *error) {
    for (CLPlacemark *placemark in placemarks) {
        NSLog(@"Placemark %@", placemark);
    }
    [geocoder release];
}];
[location release];
```

Forward Geocoding

Using an address dictionary

- Present the PeoplePicker
- Retrieve the selected address
- Geocode the address
- Format the address

Geocoding from the Address Book

Presenting the PeoplePicker

```
#include <AddressBookUI/AddressBookUI.h>
```

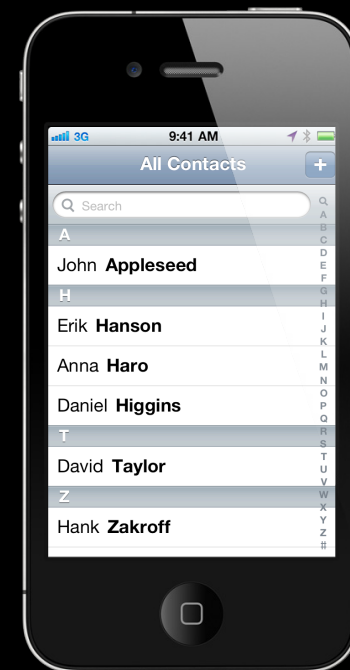
```
- (void)presentPeoplePicker
```

```
{
```

```
    ABPeoplePickerNavigationController *picker;  
    picker = [[ABPeoplePickerNavigationController alloc] init];  
    picker.peoplePickerDelegate = self;
```

```
    [self presentModalViewController:picker animated:YES];  
    [picker autorelease];
```

```
}
```



Geocoding from the Address Book

Selecting an address

```
- (BOOL)peoplePickerNavigationController:  
    (ABPeoplePickerNavigationController *)peoplePicker  
    shouldContinueAfterSelectingPerson:(ABRecordRef)person  
        property:(ABPropertyID)property  
        identifier:(ABMultiValueIdentifier)identifier  
{  
    if (property != kABPersonAddressProperty) {  
        return YES;  
    }  
  
    ABMultiValueRef value = ABRecordCopyValue(person, property);  
  
    CFIndex index = ABMultiValueGetIndexForIdentifier(value, identifier);  
    NSDictionary *address;  
    address = (NSDictionary *)ABMultiValueCopyValueAtIndex(value, index);  
    [address autorelease];  
}
```

Geocoding from the Address Book

Geocoding the address

```
{  
    ...  
  
    CLGeocoder *geocoder = [[CLGeocoder alloc] init];  
    [geocoder geocodeAddressDictionary:address  
             completionHandler:^(NSArray *placemarks,  
                                 NSError *error {  
        ...  
        [geocoder release];  
    }];  
  
    [self dismissModalViewControllerAnimated:YES];  
    return NO;  
}
```



Geocoding Data

Pierre-Luc Beaudoin
Software Engineer



Arabic Catalan Chinese Croatian Czech Danish Dutch
English Finnish French German Greek Hebrew
Hungarian Indonesian Italian Japanese Korean Malay
Norwegian Polish Portuguese Romanian Russian Spanish
Swedish Thai Turkish Ukrainian Vietnamese

English **Cupertino**

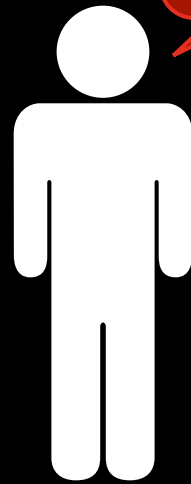
Spanish **Cupertino**

Russian **Купертино**

Korean **쿠퍼티노**

Japanese **クパチーノ**

Forward Geocoding Example



Hello!



English

4 Impasse de la fidélité
Brussels
Belgium

Forward Geocoding Example



Impasse de la fidélité 4
Bruxelles
Belgique

Forward Geocoding Example



Getrouwheidsgang 4
Brussels
België

Forward Geocoding Example

French

Impasse de la fidélité 4
Bruxelles
Belgique

Dutch

Getrouwheidsgang 4
Brussels
België

Reverse Geocoding Example

San Francisco City Hall

thoroughfare Van Ness Ave
subThoroughfare 400
subLocality Civic Center
locality Сан-Франциско
subAdministrativeArea Сан-Франциско
administrativeArea Калифорния
postalCode 94102
country Соединенные Штаты Америки



Geocoding Data

- Global coverage
- Support for multiple languages
- Results' language will match the device's locale

CLGeocoder vs. MKReverseGeocoder

- CLGeocoder:
 - Forward and reverse geocoding
 - Does not require results be displayed on a map
- MKReverseGeocoder:
 - Deprecated in iOS 5
 - MKPlacemark inherits from CLPlacemark

Summary

- Location can improve your users' experience
- Choose the right service for your use case
- Use geocoding to make location data more meaningful

More Information

Vicki Murley

Evangelist

vicki@apple.com

Documentation

Core Location Framework Reference

Location Awareness Programming Guide

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

Apple Developer Forums

<http://devforums.apple.com>

Related Sessions

Visualizing Information Geographically With Map Kit

Russian Hill
Wednesday 9:00AM

Testing Your Location-Aware Application Without Leaving Your Chair

Mission
Friday 9:00AM

Labs

Core Location Lab

Internet and Web Lab A
Tuesday 2:00PM

Map Kit Lab

Application Frameworks Lab B
Wednesday 2:00PM

Core Location Lab

Internet and Web Lab B
Thursday 4:30PM

