Visualizing Information Geographically with MapKit

Session 111

Stephane Moore
Software Engineer MapKit Team

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

What Can I Do with MapKit?

MapKit Basics

Agenda

- Map view
- Annotations
- User location
- Tracking modes
- Overlays
- Geocoding

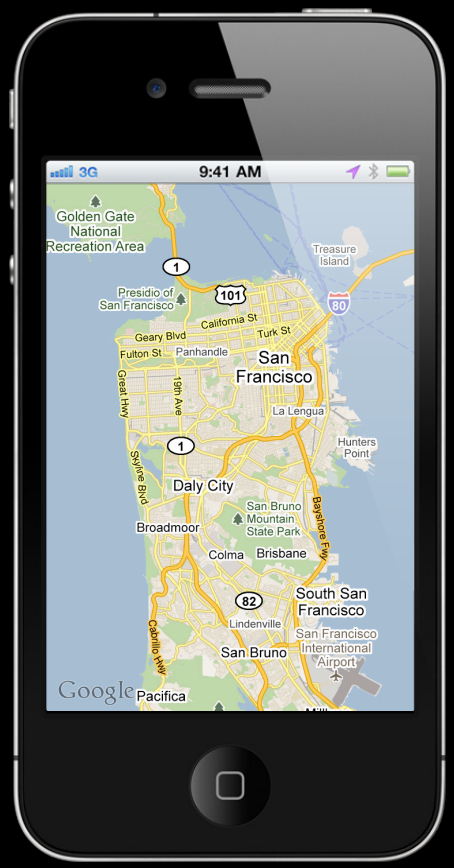
Map View


3G
9:41 AM
Golden Gate National Recreation Area
Treasure Island
Presidio of San Francisco
101
Sea Cliff
California St
Geary Blvd
Hayes Valley
San Francisco
Fulton St
17th St
280
Great Hwy
19th Ave
Hunters Point
Daly City
Skyline Blvd
San Bruno Mountain State Park
Bayshore Fwy
Broadmoor
Colma
Brisbane
Fairmont
South San Francisco
Westview
San Francisco International Airport
San Bruno
Google
Pacifica

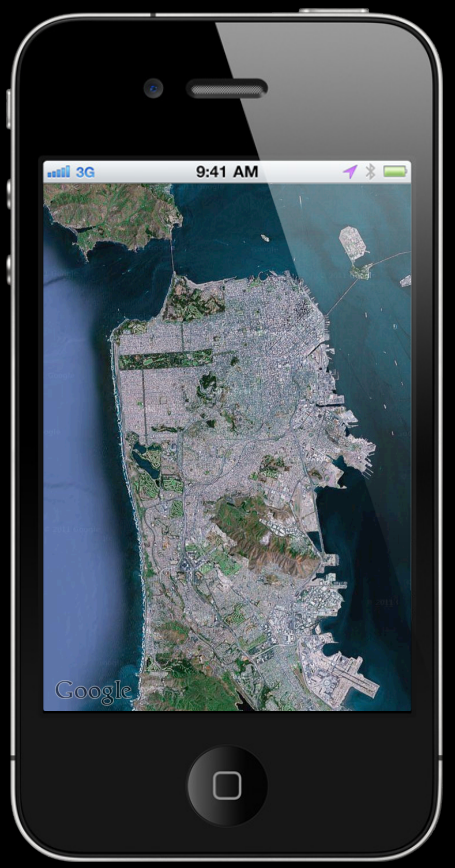
MKMapView

- UIView subclass
- Easy to use
- High performance

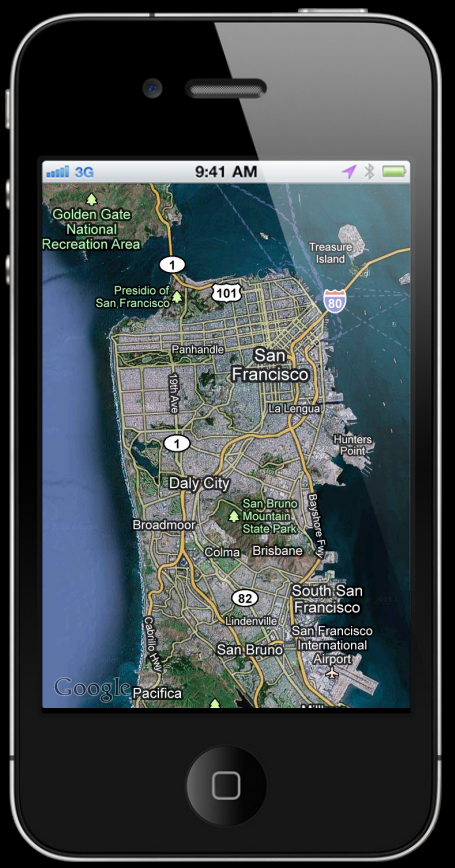
Map Types



Regular

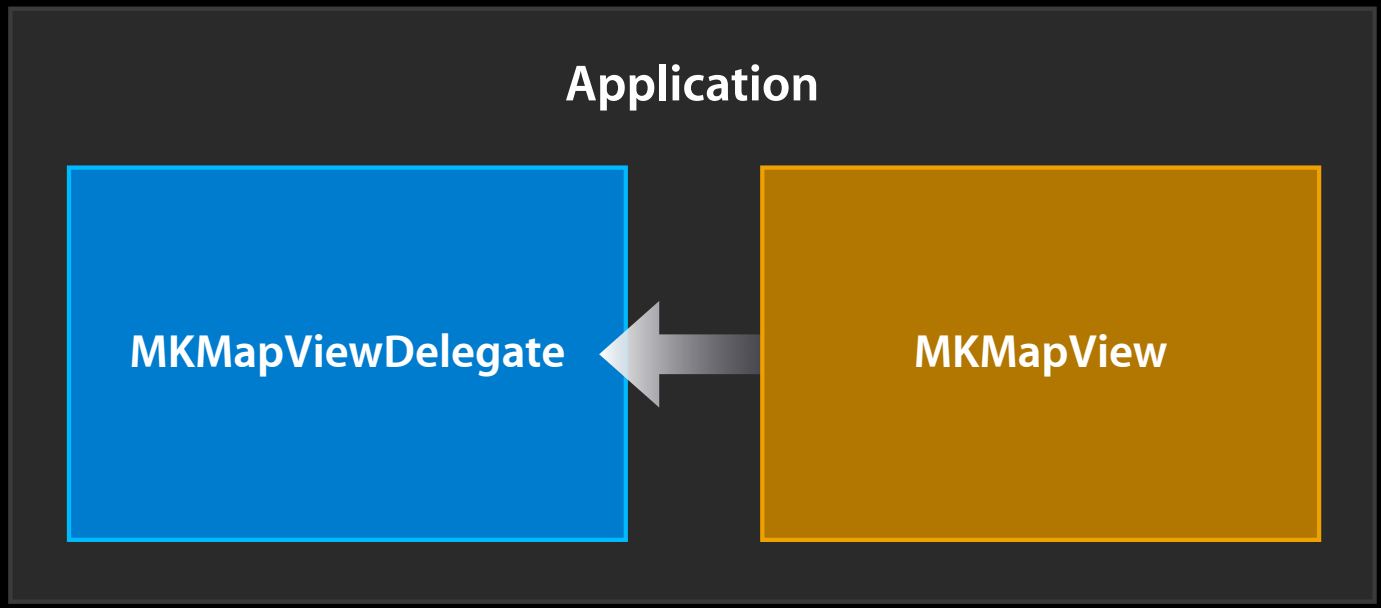


Satellite

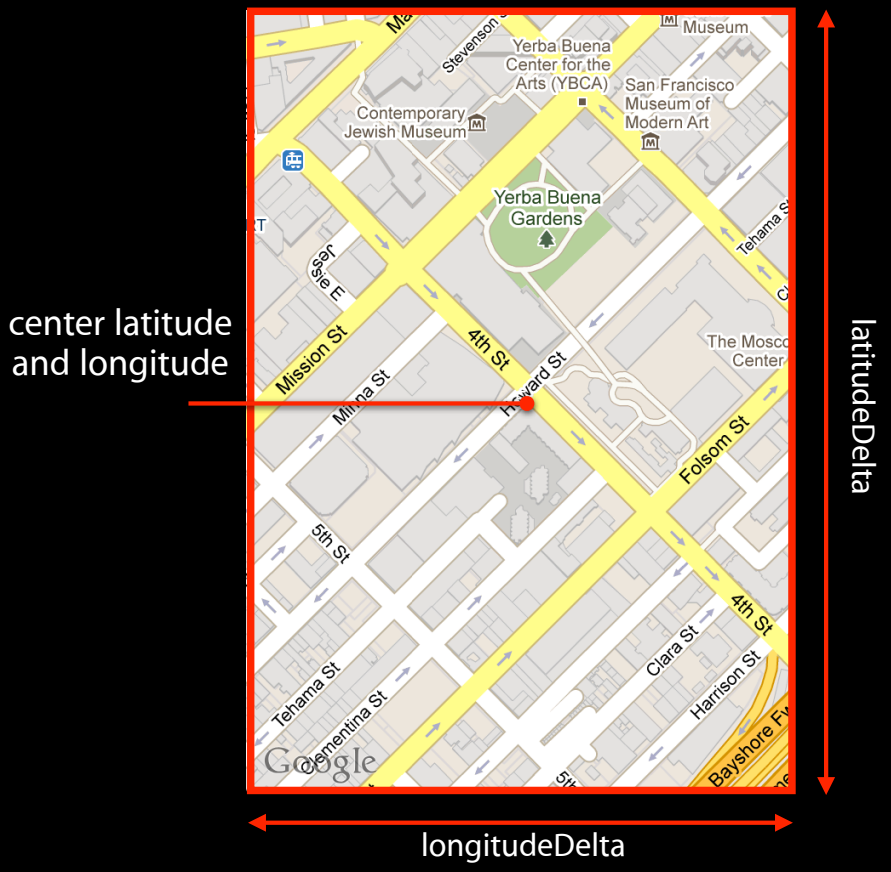


Hybrid

MKMapViewDelegate



Map Regions



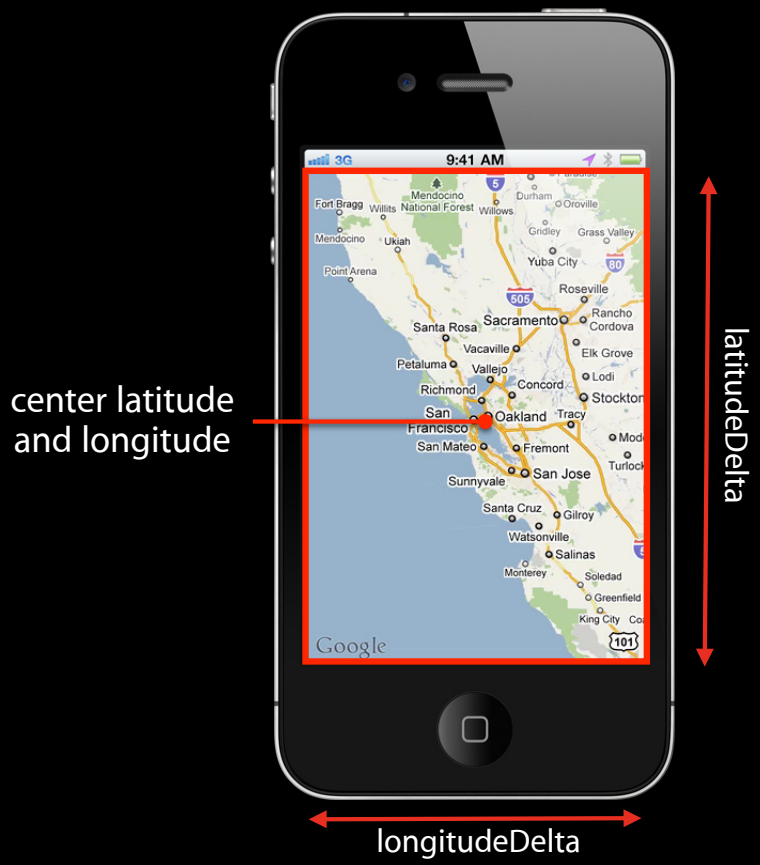

```
typedef struct {
  CLLocationDegrees latitude;
  CLLocationDegrees longitude;
} CLLocationCoordinate2D;

typedef struct {
  CLLocationDegrees latitudeDelta;
  CLLocationDegrees longitudeDelta;
} MKCoordinateSpan;

typedef struct {
  CLLocationCoordinate2D center;
  MKCoordinateSpan span;
} MKCoordinateRegion;
```

Positioning the Map

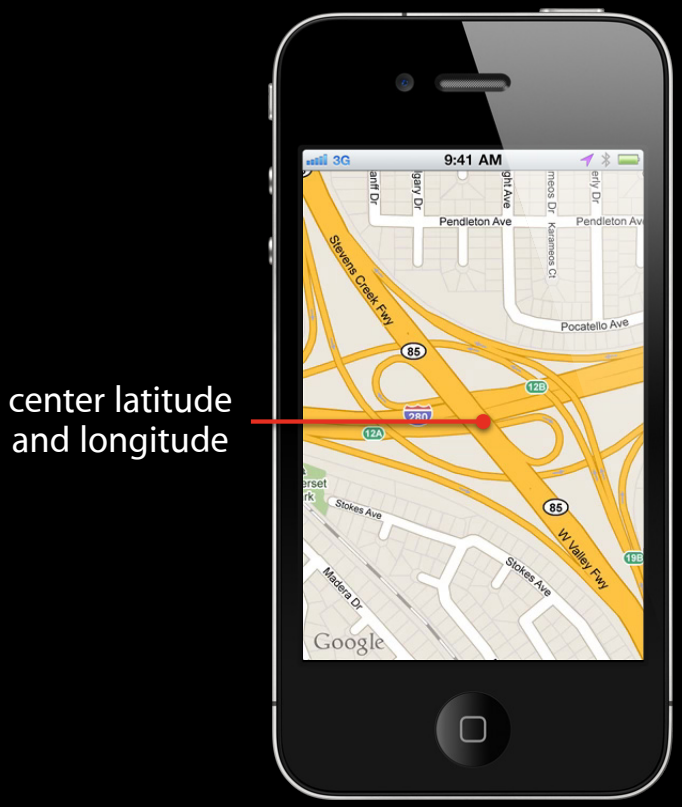
Map regions



- Use the region property to control the map position and scale
- The map view will find the optimal display region
- Animated transitions

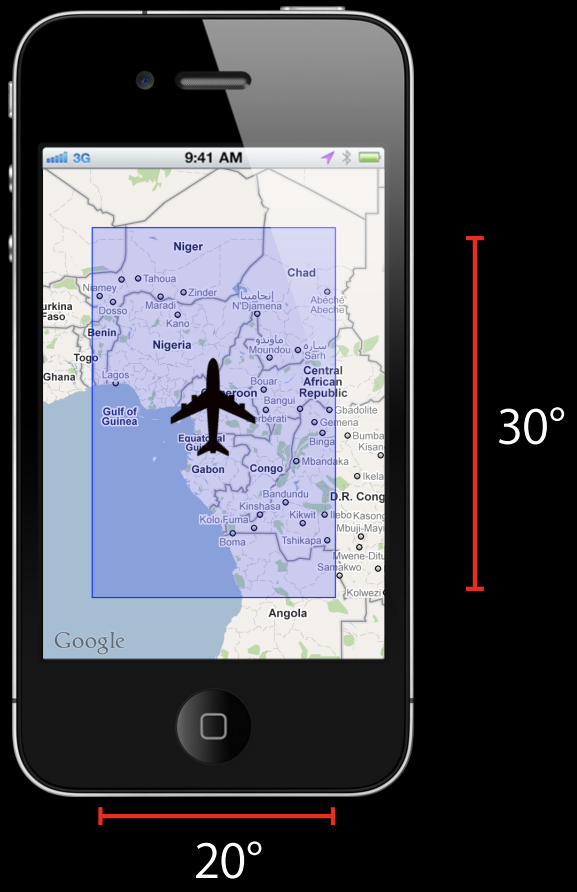
Positioning the Map

Center coordinates



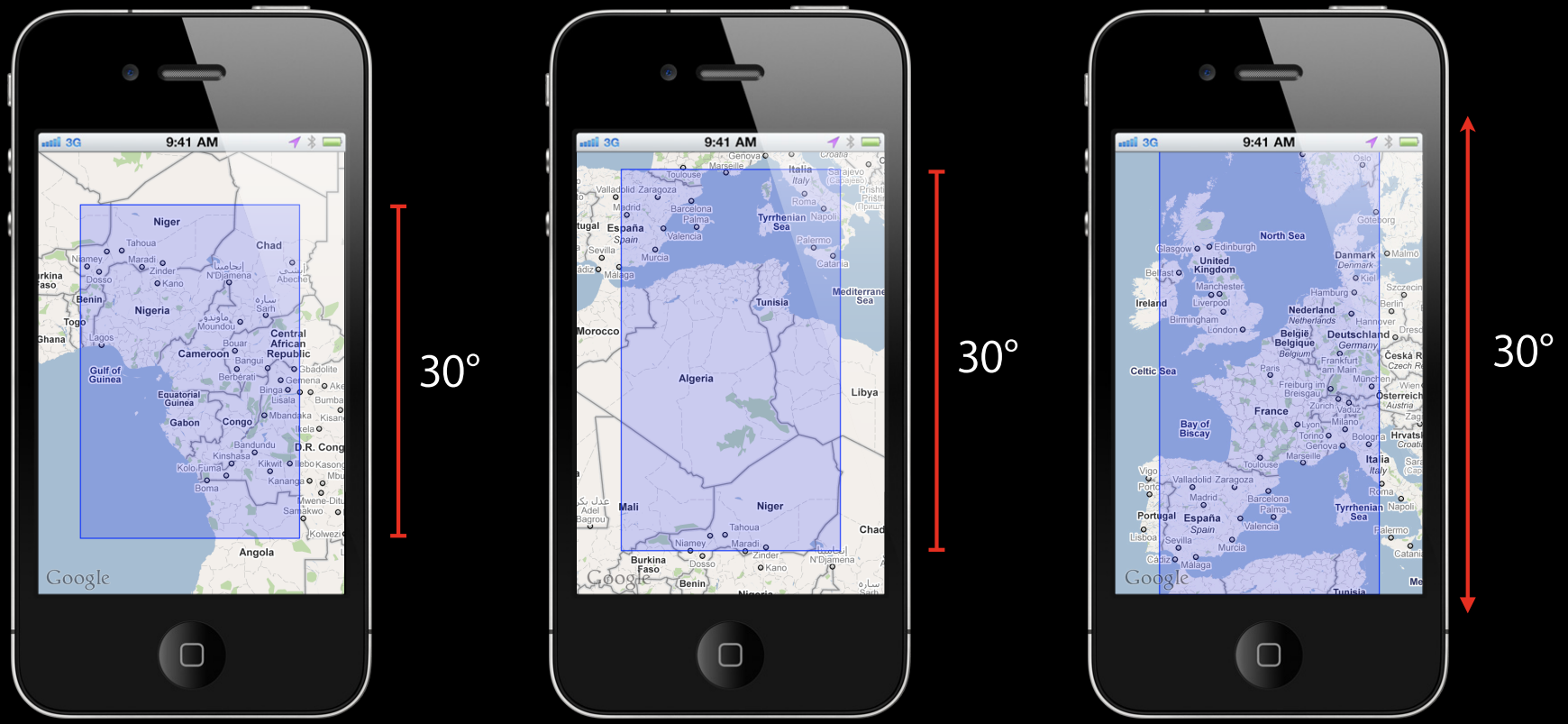
- Use centerCoordinate to control map position without changing scale

Map Region Pitfalls



Map Region Pitfalls

Slide 13
Fixed build

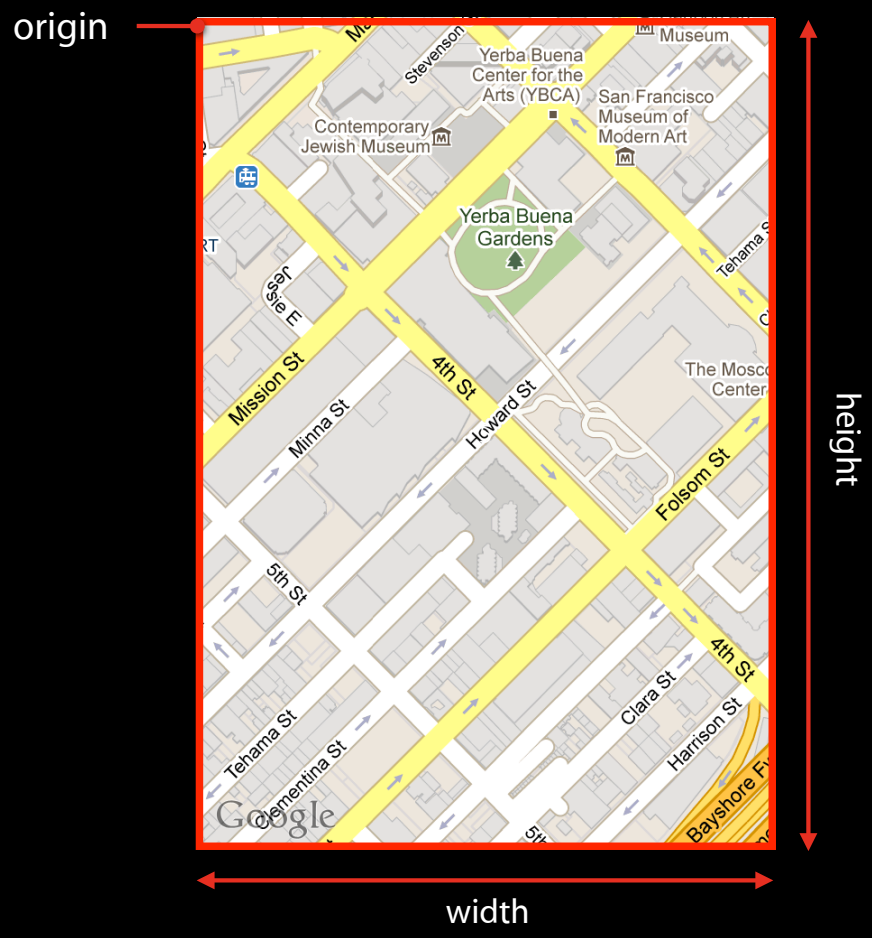


Mercator Projection

MKMapPoint

- Can represent any point on the map
- Linearly proportional to screen points
- Use MKMapPointForCoordinate to convert from latitude/longitude to MKMapPoints

Map Rects




```
typedef struct {
  double x;
  double y;
} MKMapPoint;
```

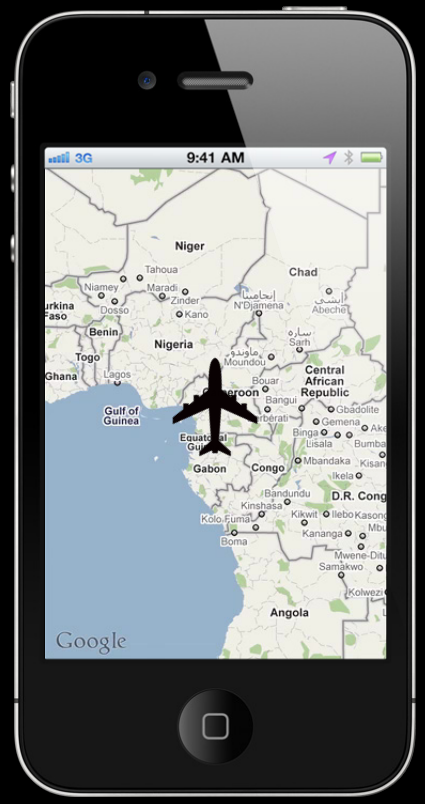
```
typedef struct {
  double width;
  double height;
} MKMapSize;
```

```
typedef struct {
  MKMapPoint origin;
  MKMapSize size;
} MKMapRect;
```

Map Rects

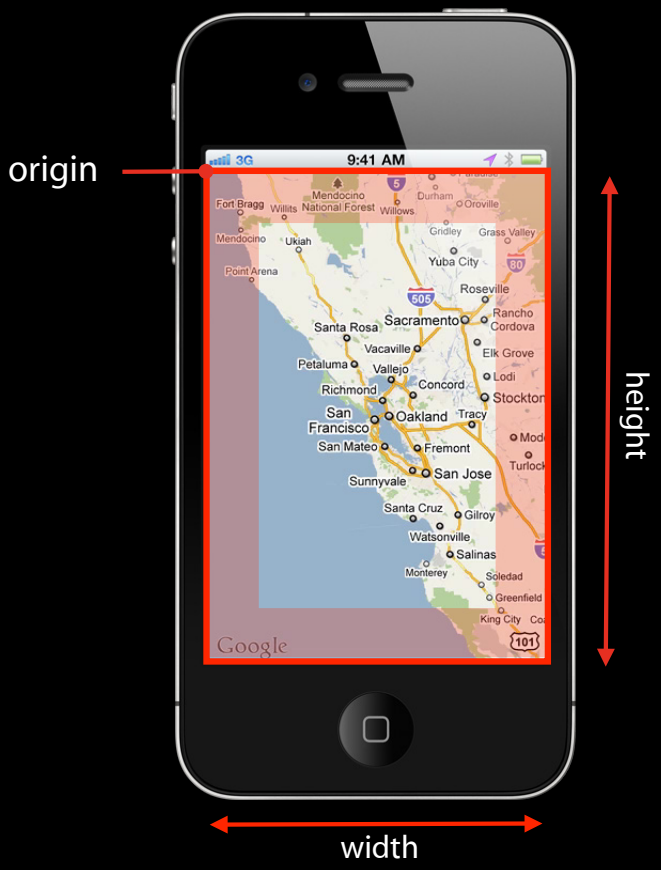

3G
9:41 AM
Niger
Chad
Nigeria
Benin
Togo
Ghana
Cameroon
Central African Republic
Gulf of Guinea
Equatorial Guinea
Gabon
Congo
Angola
Google
3G
9:41 AM
Tunisia
Algeria
Libya
Morocco
Mali
Niger
Burkina Faso
Benin
Google
3G
9:41 AM
North Sea
United Kingdom
Ireland
Nederland
Belgique
Deutschland
France
Bay of Biscay
Celtic Sea
Portugal
España
Italia
Tyrrhenian Sea
Google

Map Rects


3G
9:41 AM
Niger
Tahoua
Chad
Niamey
Maradi
Zinder
Dosso
Kano
Benin
Nigeria
Togo
Ghana
Lagos
Moundou
Bouar
Central African Republic
Bangui
Gulf of Guinea
Gabon
Congo
Gemena
Binga
Mbandaka
Ikela
Bandundu
Kinshasa
Kikwit
Kananga
Boma
Angola
Google

Positioning the Map

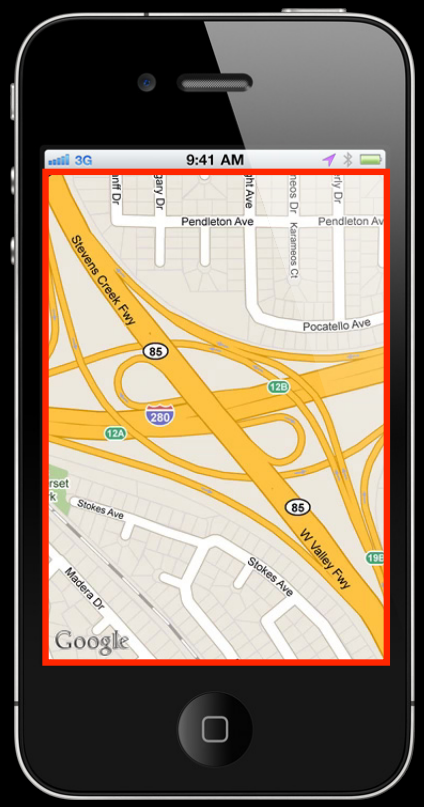
Map rects



- Use visibleMapRect property to change the map position and scale
- The map view finds the optimal display region
- You can supply edgePaddings to get padding around your viewing area

User Interaction

Capturing region changes

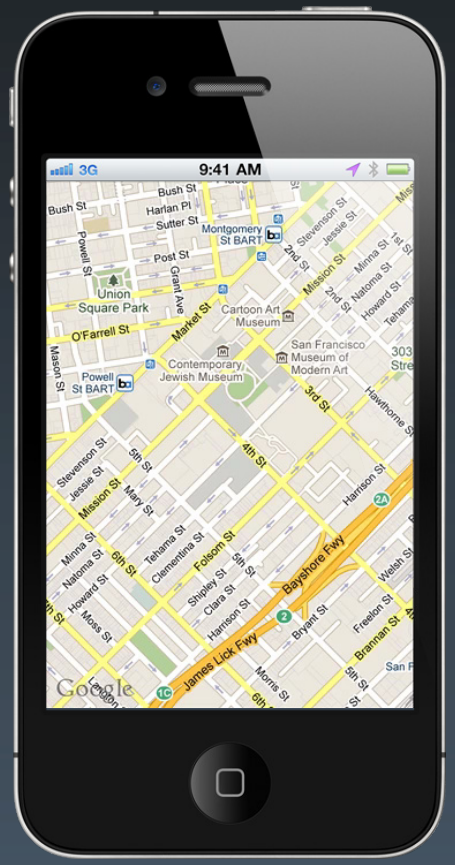


MKMapViewDelegate

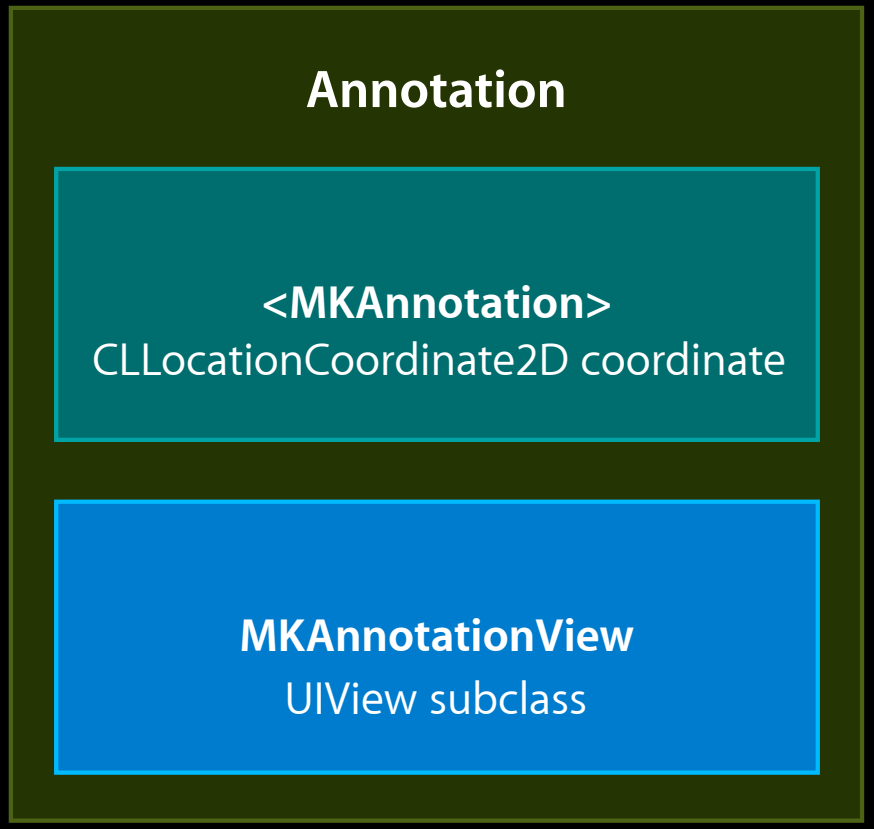
mapView:regionWillChangeAnimated:

mapView:regionDidChangeAnimated:

Annotations



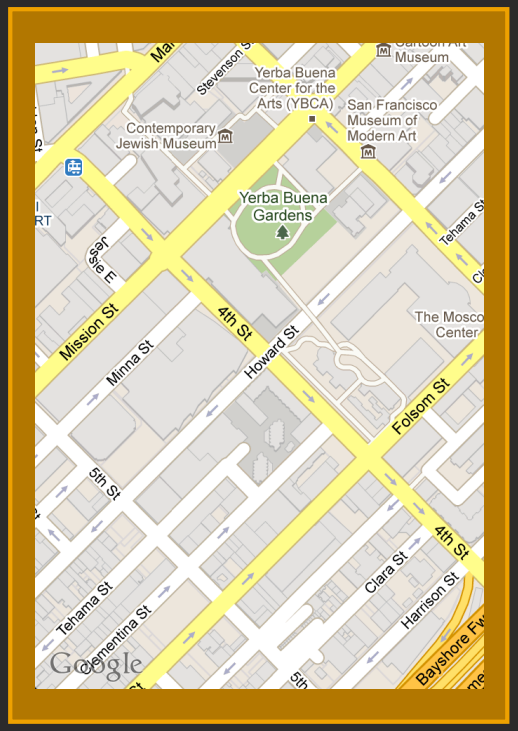
What Is an Annotation?



Annotations Flow

Step 1—Annotation added to the map

Application

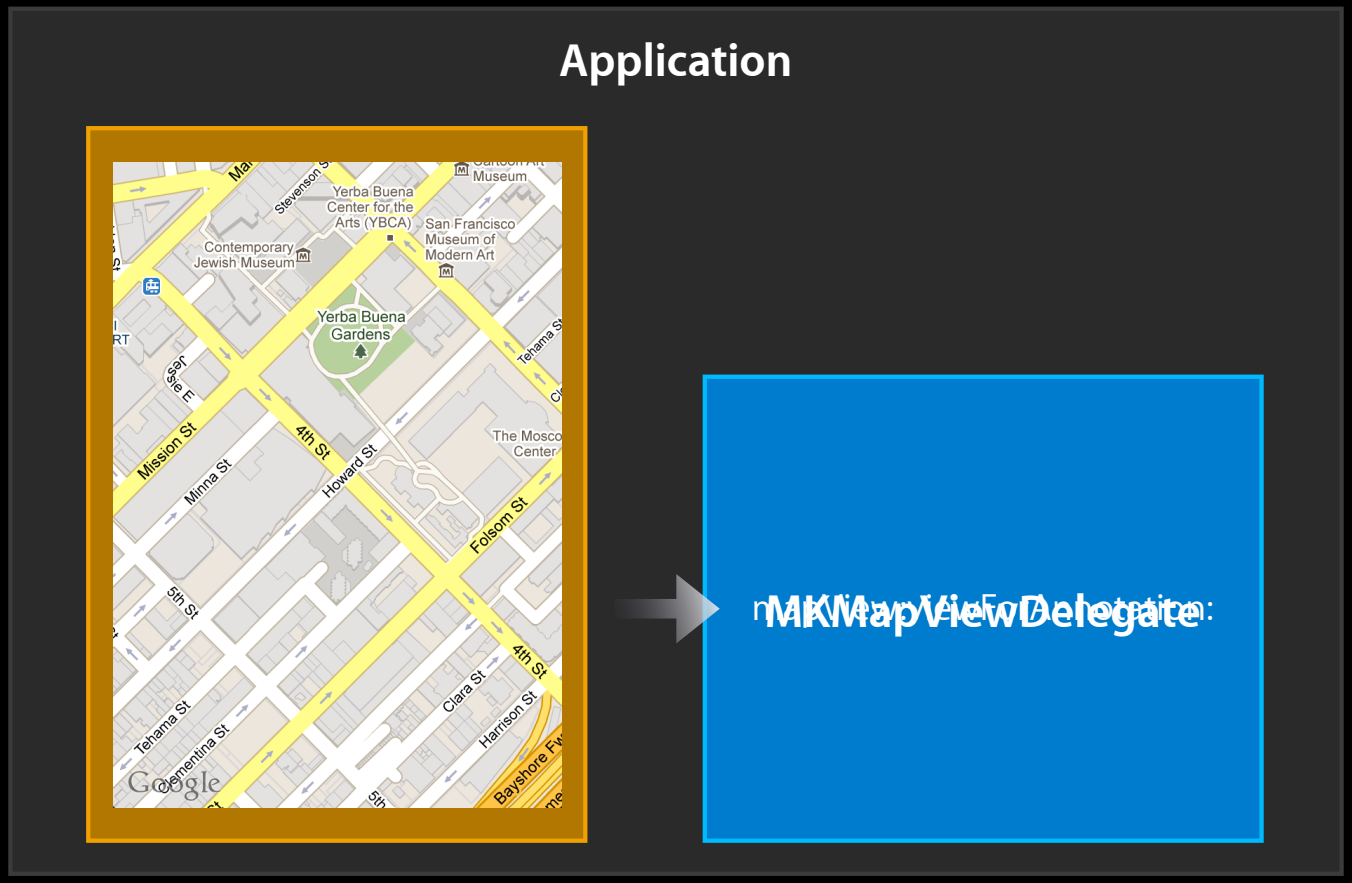


`37.78326, -122.40270`

MKMapViewDelegate

Annotations Flow

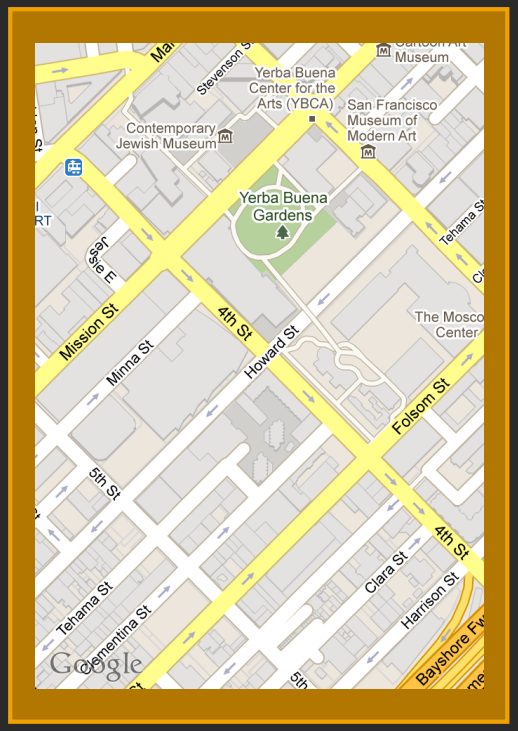
Step 2—Annotation view requested



Annotations Flow

Step 3—Annotation view created

Application



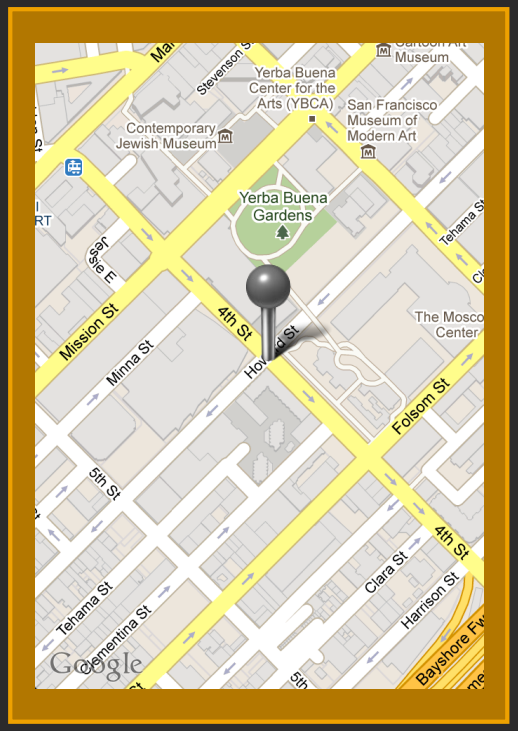
MKMapViewDelegate

mapView:v Annotation:

Annotations Flow

Step 4—Annotation view positioned

Application

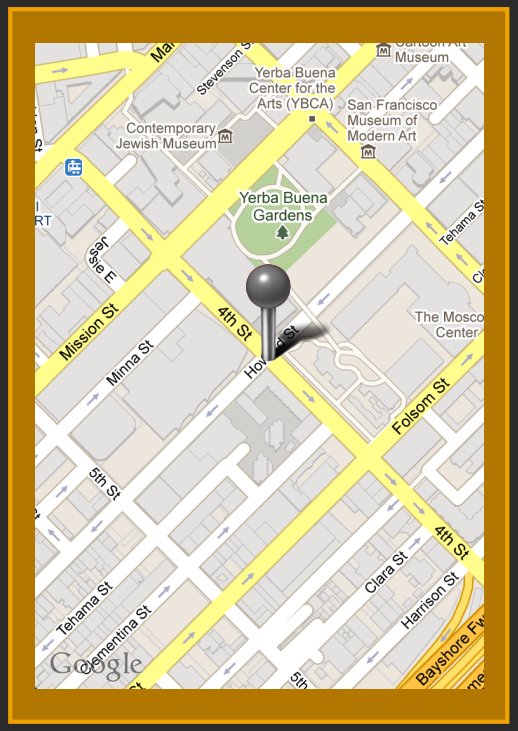


MKMapViewDelegate

Annotations Flow

Step 5—Annotation view added

Application

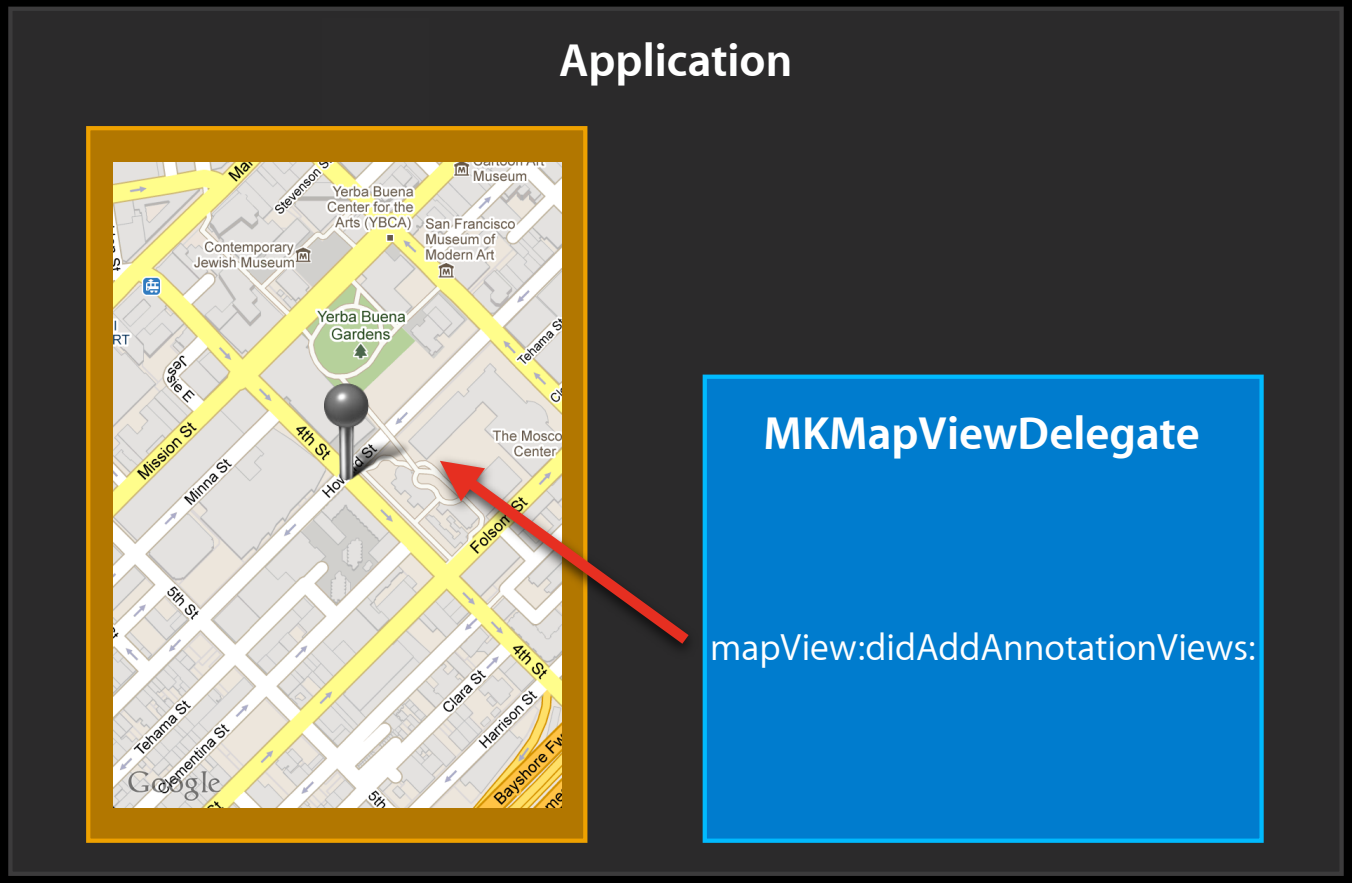


MKMapViewDelegate

mapView:didAddAnnotationViews:

Annotations Flow

Step 6—Delegate optionally animates annotations



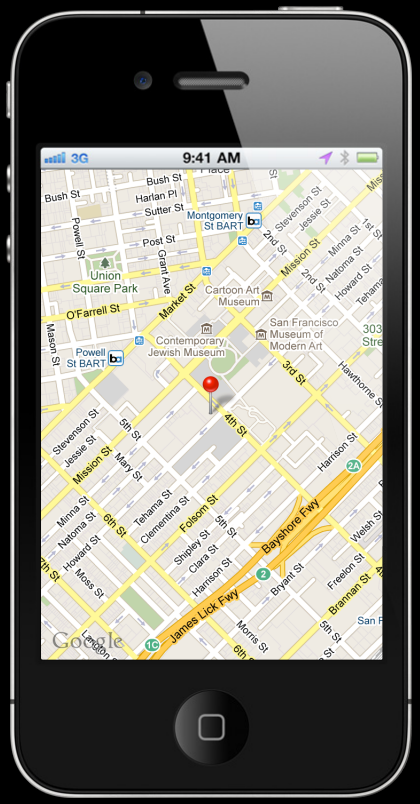
Animating Annotations

```
- (void)mapView:(MKMapView *)mapView
didAddAnnotationViews:(NSArray *)annotationViews
{
    for (MKAnnotationView *view in annotationViews)
    {
        CGRect endFrame = view.frame;

        CGRect startFrame = endFrame;
        startFrame.origin.x = view.center.x;
        startFrame.origin.y += endFrame.size.height;
        startFrame.size.width = 0;
        startFrame.size.height = 0;
        view.frame = startFrame;

        [UIView beginAnimations:nil context:nil];
        [UIView setAnimationDuration:1.0];
        view.frame = endFrame;
        [UIView commitAnimations];
    }
}
```

Selecting Annotations



MKMapViewDelegate

mapView:didSelectAnnotationView:

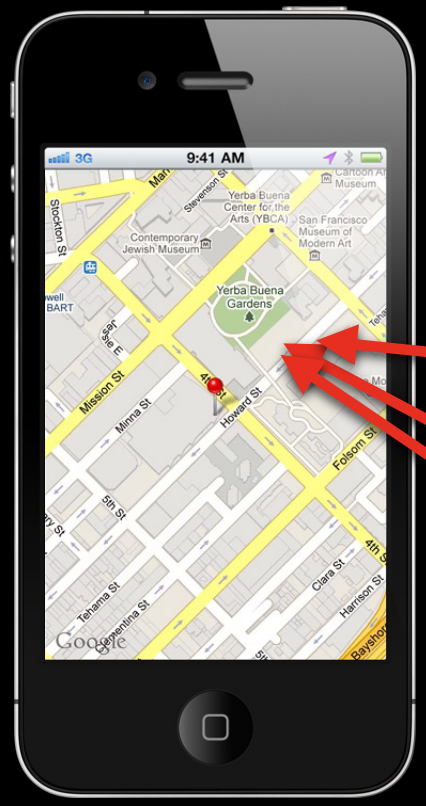
mapView:didDeselectAnnotationView:

MKMapView

selectAnnotation:animated:

deselectAnnotation:animated:

Showing Callouts



Annotation

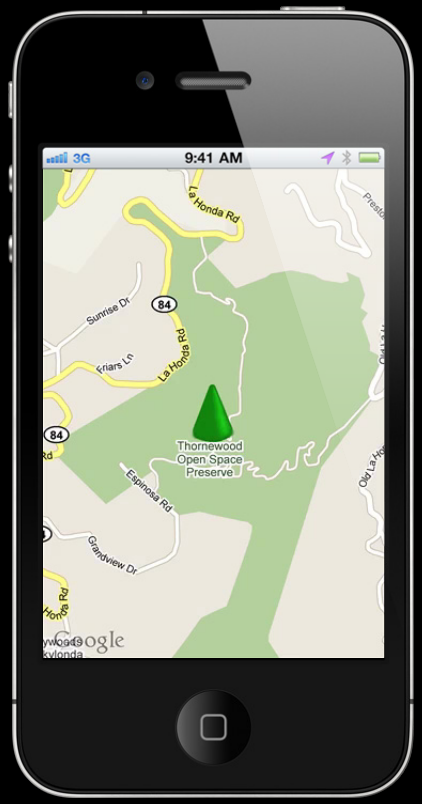
MKAnnotationView

- Set canShowCallout
- Callout accessories

<MKAnnotation>

- Title
- Subtitle

Custom Selection UI



Annotation

MKAnnotationView

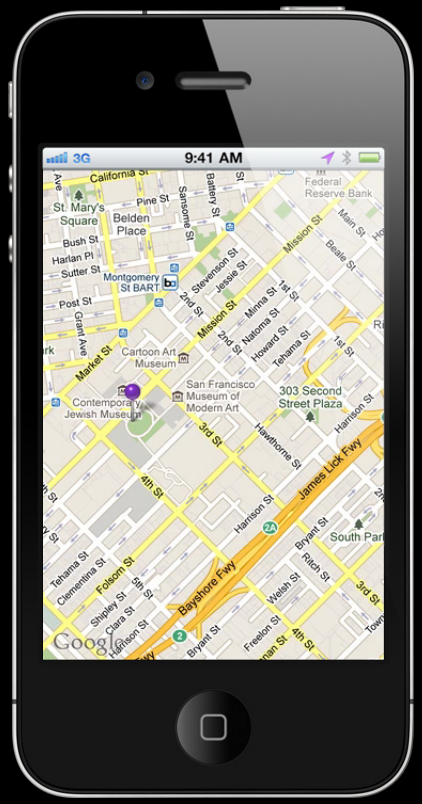
- setSelected:animated:

MKMapViewDelegate

mapView:didSelectAnnotationView:

mapView:didDeselectAnnotationView:

Draggable Annotations



Annotation

MKAnnotationView

- Set draggable

<MKAnnotation>

- Implement setCoordinate:

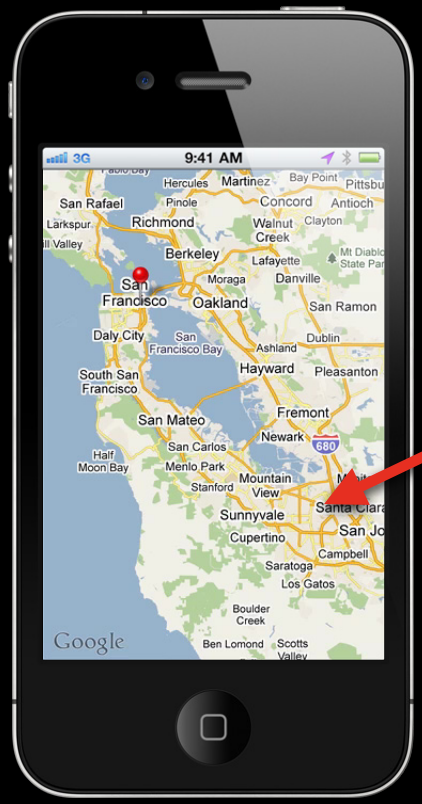
Animating Draggable Annotations

Annotation

MKAnnotationView

- setDragState:animated:

Property Animation



Annotation

<MKAnnotation>

- coordinate
- title
- subtitle

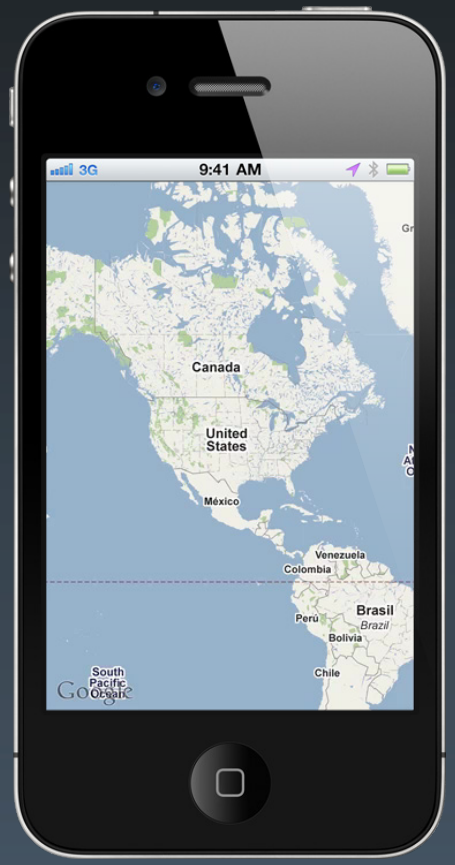
MKAnnotationView

- Callout accessories

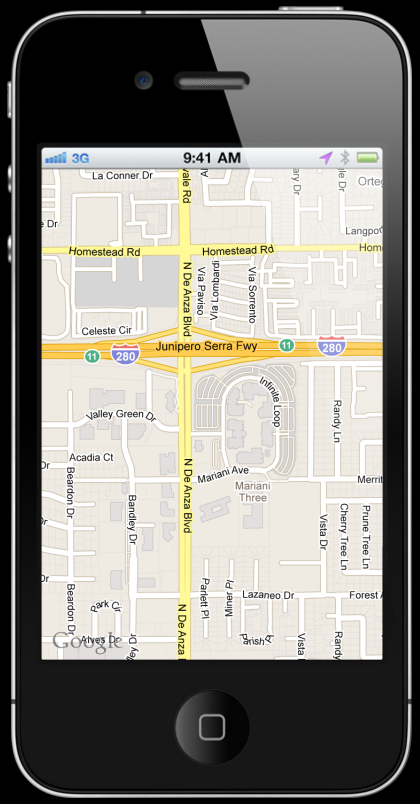
Demo

Matt Jarjoura
Software Engineer

User Location



Showing User Location on a Map



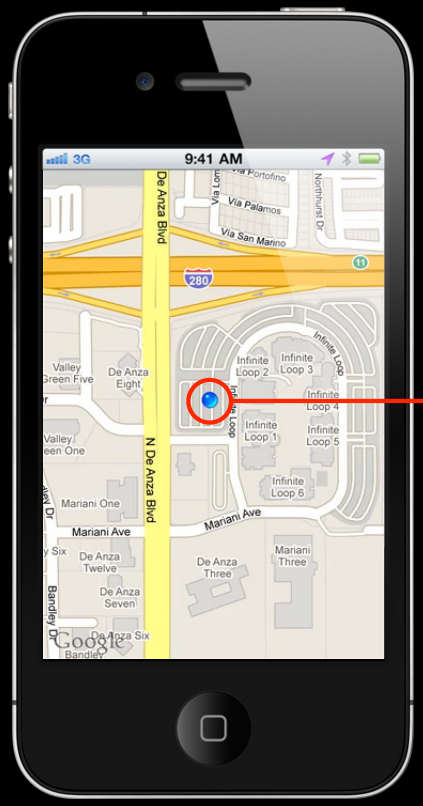
- Set showsUserLocation
- Requires user authorization

MKMapViewDelegate

mapView:didUpdateUserLocation:

mapView:didFailToLocateUserWithError:

Things to Know About User Location

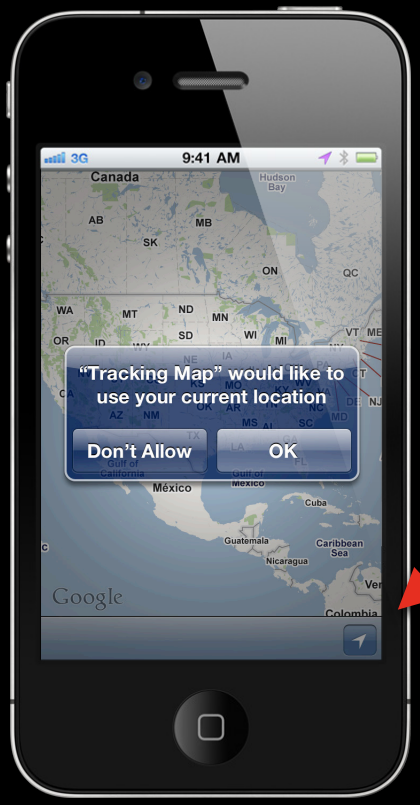


userLocationVisible

animation

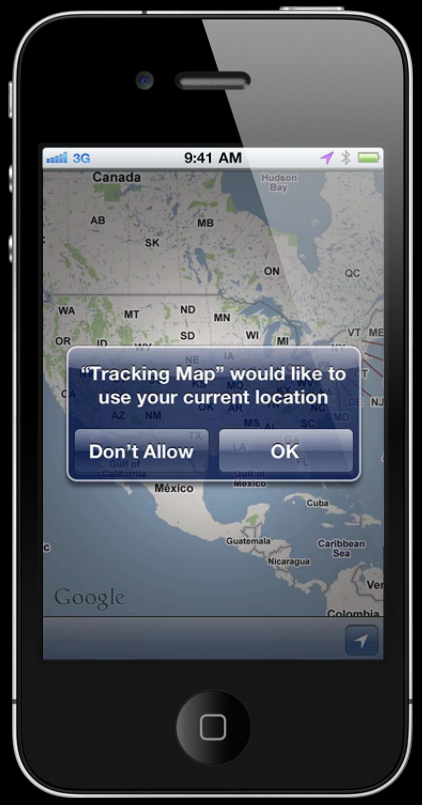
Tracking Modes

Tracking Modes



- Set userTrackingMode on MKMapView
- Requires user authorization
- MKUserTrackingBarButtonItem

Tracking Modes

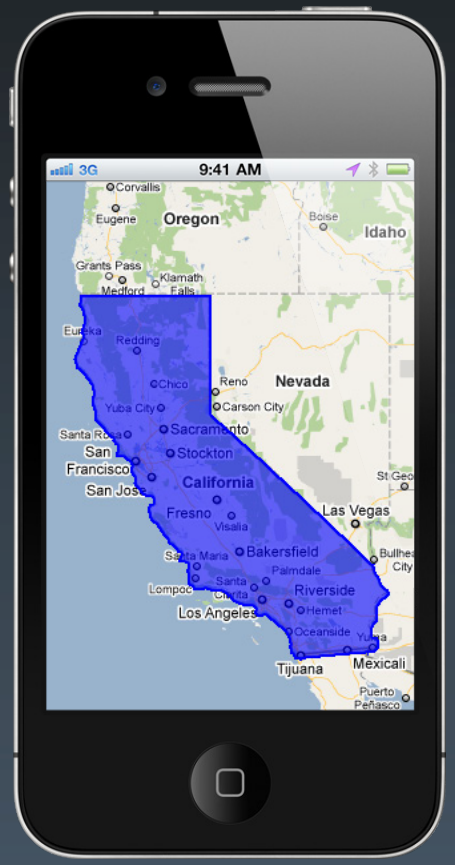


Supported tracking modes

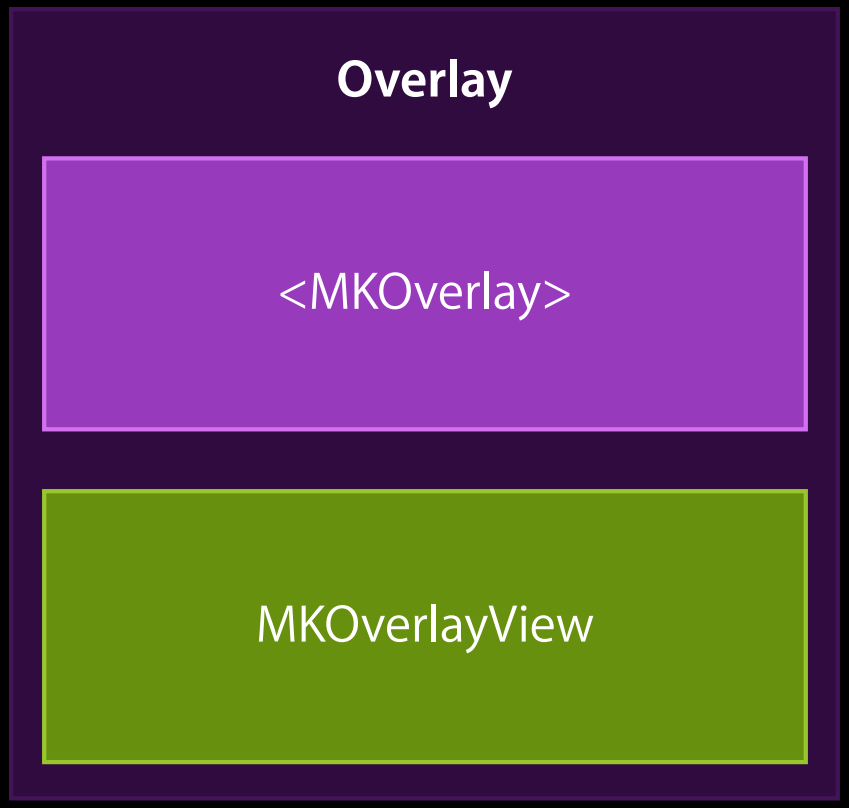
MKUserTrackingModeFollow

MKUserTrackingModeFollowWithHeading

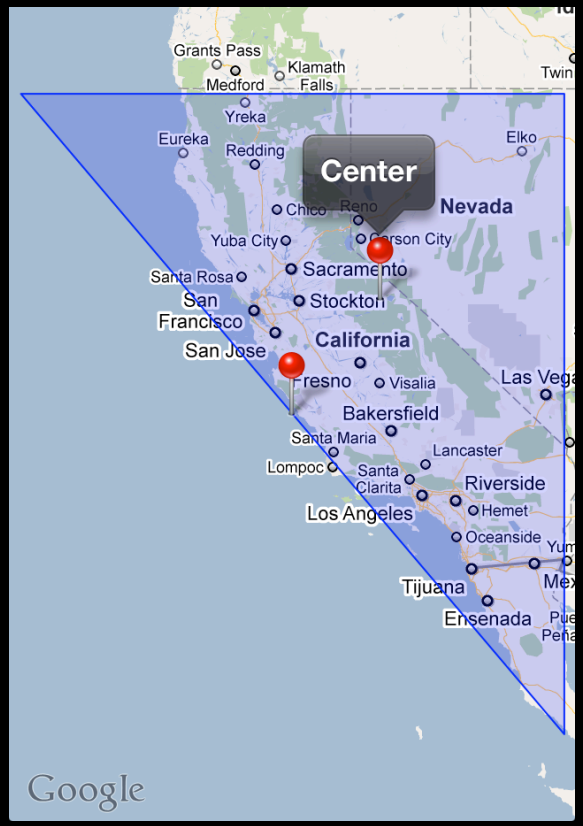
Overlays


3G
9:41 AM
Corvallis
Eugene
Oregon
Boise
Idaho
Grants Pass
Medford
Klamath Falls
Eureka
Redding
Chico
Reno
Nevada
Yuba City
Carson City
Sacramento
Santa Rosa
San Francisco
Stockton
San Jose
California
Fresno
Visalia
Las Vegas
Bakersfield
Palmdale
Riverside
Hemet
Los Angeles
Oceanside
Tijuana
Mexicali
Puerto Peñasco

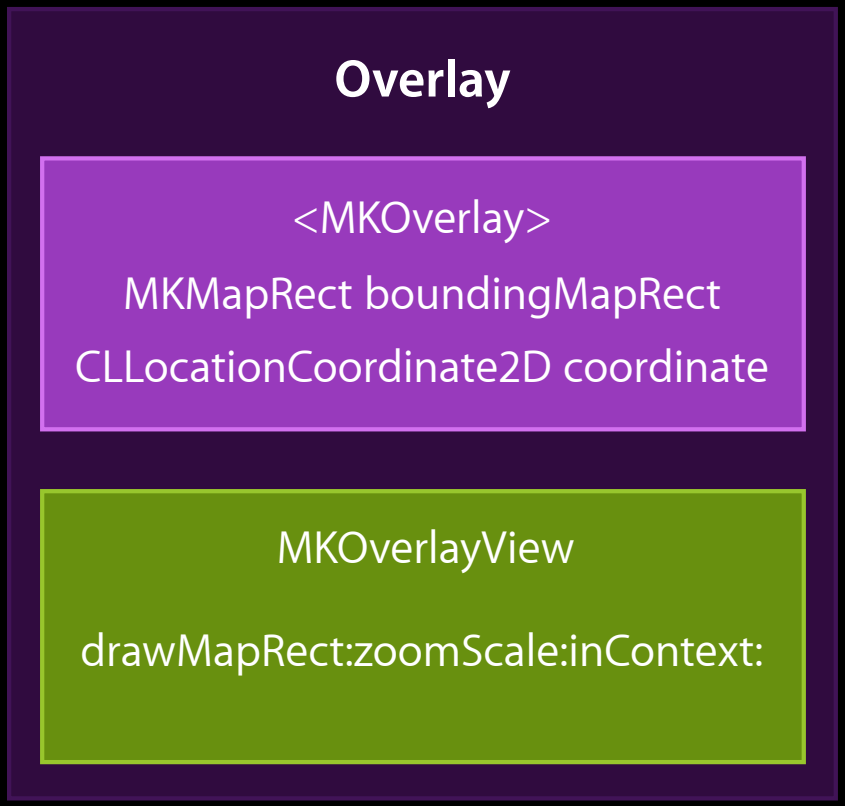
What Is an Overlay?



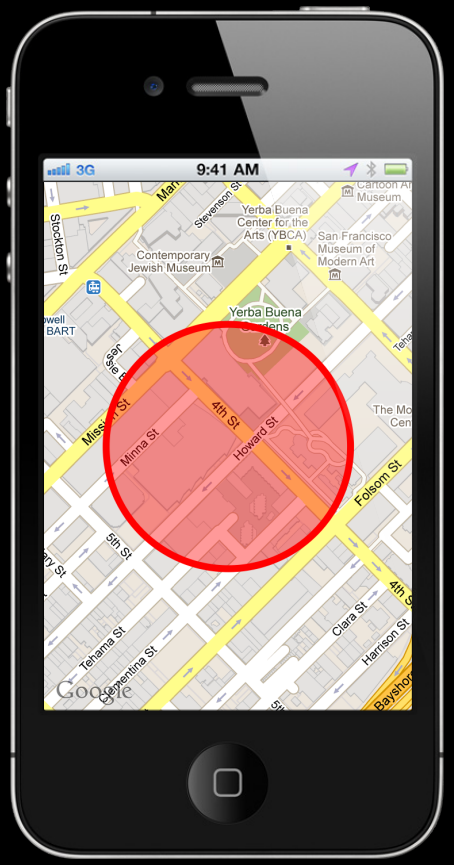
What Is an Overlay?



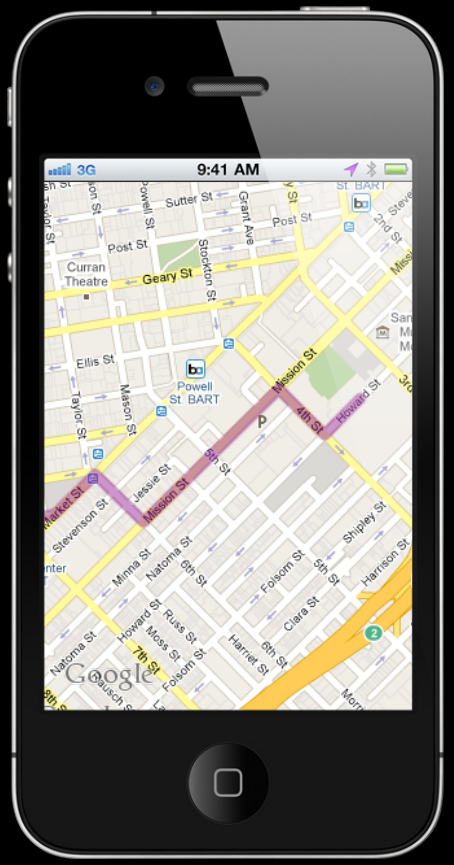
What Is an Overlay?



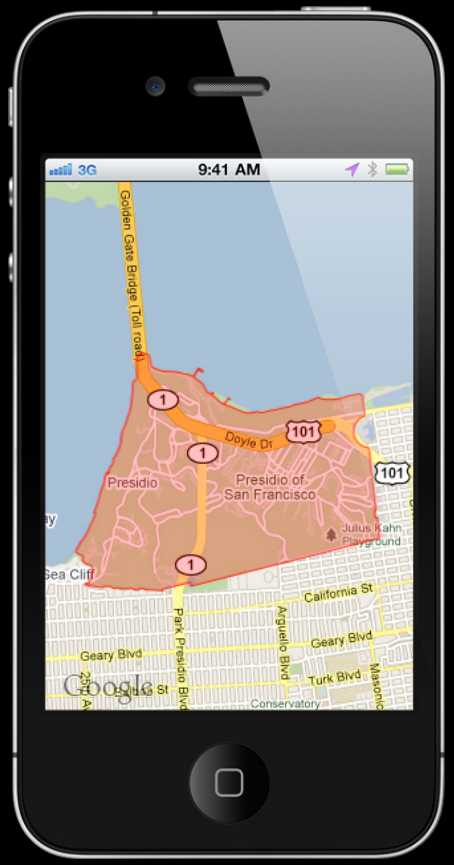
MapKit Overlays



MKCircle



MKPolyline



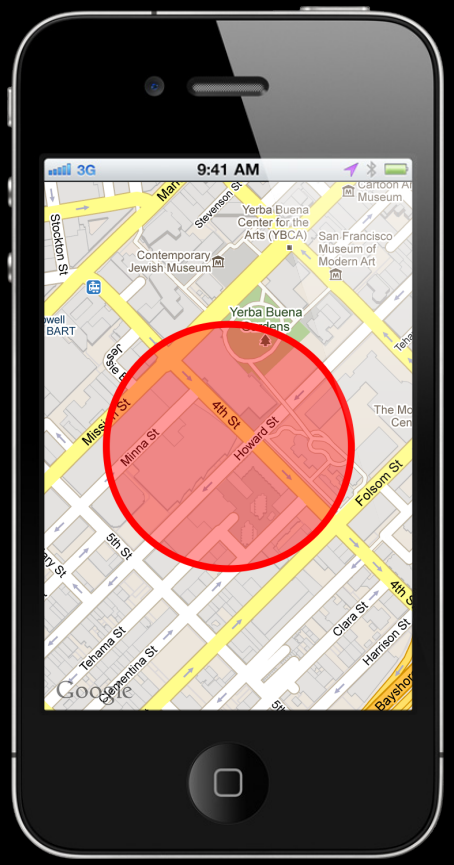
MKPolygon

Adding Overlays to the Map View

```
- (void)viewDidLoad
{
    CLLocationCoordinate2D center = CLLocationCoordinate2DMake(37.78326, -122.4027);
    MKCircle *circle = [MKCircle circleWithCenterCoordinate:center radius:200];
    [mapView addOverlay:circle];
}


- (MKOverlayView *)mapView:(MKMapView *)map viewForOverlay:(id <MKOverlay>)overlay
{
    MKCircleView *circleView = [[MKCircleView alloc] initWithOverlay:overlay];
    circleView.strokeColor = [UIColor redColor];
    circleView.fillColor = [[UIColor redColor] colorWithAlphaComponent:0.4];
    return [circleView autorelease];
}
```

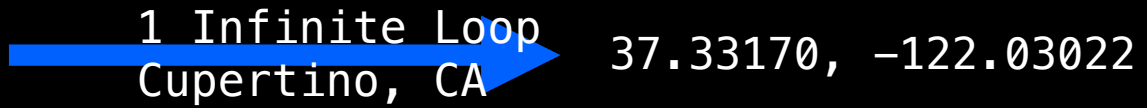
Adding Overlays to the Map View



Geocoding

What Is Forward Geocoding?

Turns an address into a coordinate



What Is Forward Geocoding?

Turns an address into a coordinate

What Is Reverse Geocoding?

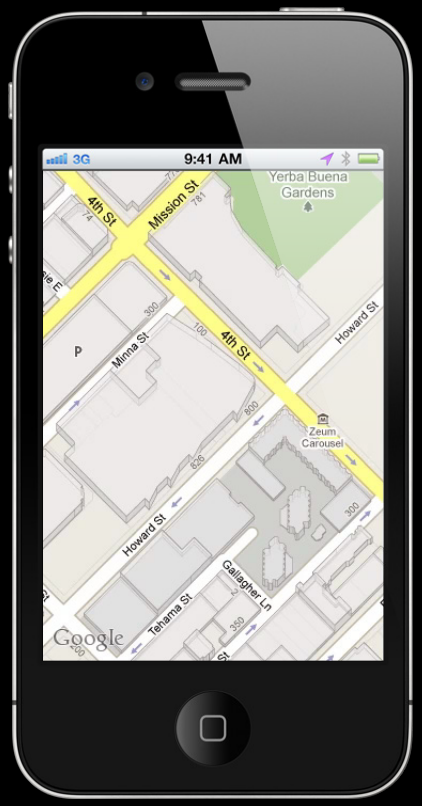
Turns a coordinate into an address

37.33170, –122.03022

1 Infinite Loop
Cupertino, CA

What Is Reverse Geocoding?

Turns a coordinate into an address




MKReverGeocoder

MKReverseGeocoder

CLGeocoder

CLGeocoder

- Forward geocoding

```
geocodeAddressString:completionHandler:
geocodeAddressDictionary:completionHandler:
```

- Reverse geocoding

```
reverseGeocodeLocation:completionHandler:
```

Using CLGeocoder

- Completion handler

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

- Array of CLPlacemarks on success
- NSError on failure

CLPlacemark

location	37.77571, -122.41946
subThoroughfare	400
thoroughfare	Van Ness Ave
subLocality	Civic Center
locality	San Francisco
subAdministrativeArea	San Francisco
administrativeArea	California
postalCode	94102
country	United States

MKPlacemark

- Now a subclass of CLPlacemark
  - Supports all of the existing properties and more
- Create MKPlacemarks from CLPlacemarks

```
MKPlacemark *placemark = [[MKPlacemark alloc] initWithPlacemark:clPlacemark];
```

- It is an annotation and can be added to map views
- Dictionary representation can be used with Address Book

Reverse Geocoding


```
CLLocation *location = [[CLLocation alloc] initWithLatitude:37.33170
                                                  longitude:-122.03022];

CLGeocoder *geocoder = [[CLGeocoder alloc] init];
[geocoder reverseGeocodeLocation: location
               completionHandler: ^(NSArray *placemarks, NSError *error) {

    // Check for returned placemarks
    if (placemarks && placemarks.count > 0) {
        CLPlacemark *topResult = [placemarks objectAtIndex:0];

        // Create an MKPlacemark and add it to the map view
        MKPlacemark *placemark = [[MKPlacemark alloc] initWithPlacemark:topResult];
        [mapView addAnnotation:placemark];
        [placemark release];
    }
    [geocoder release];
}];

[location release];
```

Forward Geocoding Address Strings


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

CLGeocoder *geocoder = [[CLGeocoder alloc] init];
[geocoder geocodeAddressString: addressString
            completionHandler: ^(NSArray *placemarks, NSError *error) {
    // Check for returned placemarks
    if (placemarks && placemarks.count > 0) {
        CLPlacemark *topResult = [placemarks objectAtIndex:0];

        // Create an MKPlacemark and add it to the map view
        MKPlacemark *placemark = [[MKPlacemark alloc] initWithPlacemark:topResult];
        [mapView addAnnotation:placemark];
        [placemark release];
    }
    [geocoder release];
}];
```

Forward Geocoding Address Dictionaries


```
NSDictionary *addressDictionary = [[NSDictionary alloc] initWithObjectsAndKeys:
        @"1 Infinite Loop", kABPersonAddressStreetKey,
        @"Cupertino",       kABPersonAddressCityKey,
        @"CA",              kABPersonAddressStateKey
    ];

CLGeocoder *geocoder = [[CLGeocoder alloc] init];
[geocoder geocodeAddressDictionary: addressDictionary
                 completionHandler: ^(NSArray *placemarks, NSError *error) {
    // Check for returned placemarks
    if (placemarks && placemarks.count > 0) {
        CLPlacemark *topResult = [placemarks objectAtIndex:0];

        // Create an MKPlacemark and add it to the map view
        MKPlacemark *placemark = [[MKPlacemark alloc] initWithPlacemark:topResult];
        [mapView addAnnotation:placemark];
        [placemark release];
    }
    [geocoder release];
}];
[addressDictionary release];
```

Demo

Matt Jarjoura
Software Engineer

Summary

- MapKit lets you present rich annotated maps with powerful contextual information
- Tracking modes provide new ways of displaying user-centric data
- Overlays provide flexible ways of showing spatial data
- Adopt new geocoding APIs for a better user experience

Related Sessions

What's New in Core Location	Pacific Heights Tuesday 10:15AM
Testing Your Location-Aware Application Without Leaving Your Chair	Mission Friday 9:00AM

Labs

MapKit Lab	Application Frameworks Lab B Wednesday 2:00-6:00PM
Core Location Lab	Internet and Web Lab B Thursday 4:30-5:30PM

More Information

Bill Dudney
Application Frameworks Evangelist
dudney@apple.com

Documentation
MapKit Framework Reference
http://developer.apple.com/library/ios/navigation

Core Location Framework Reference
http://developer.apple.com/library/ios/navigation

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


WWDC2011