

Getting Around with Map Kit

Session 300 and 310

Brady Law

Software Engineer - Maps Client Team

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

Agenda



Agenda



- The new look of Map Kit

Agenda



- The new look of Map Kit
- Launching Maps with MKMapItem

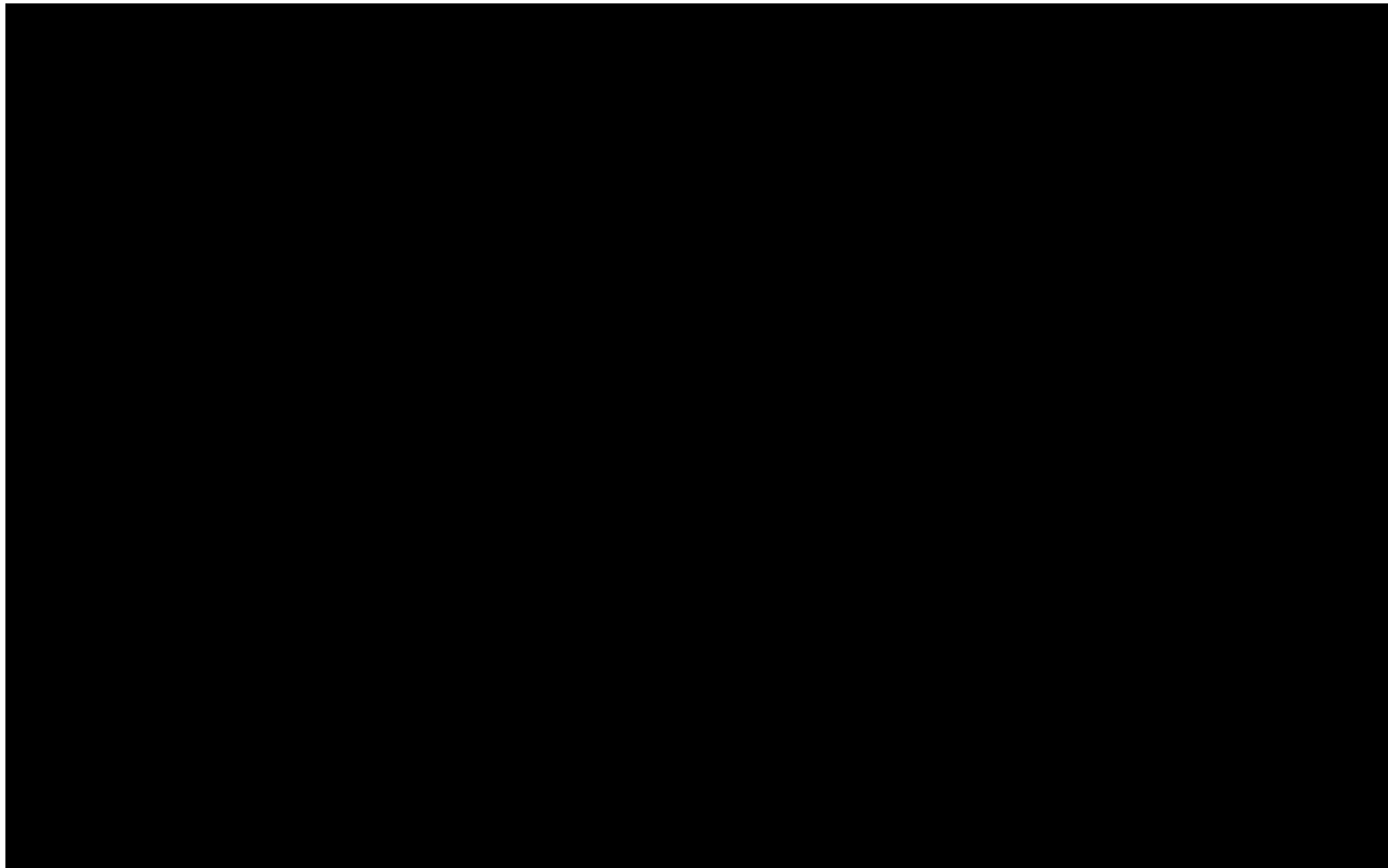
Agenda

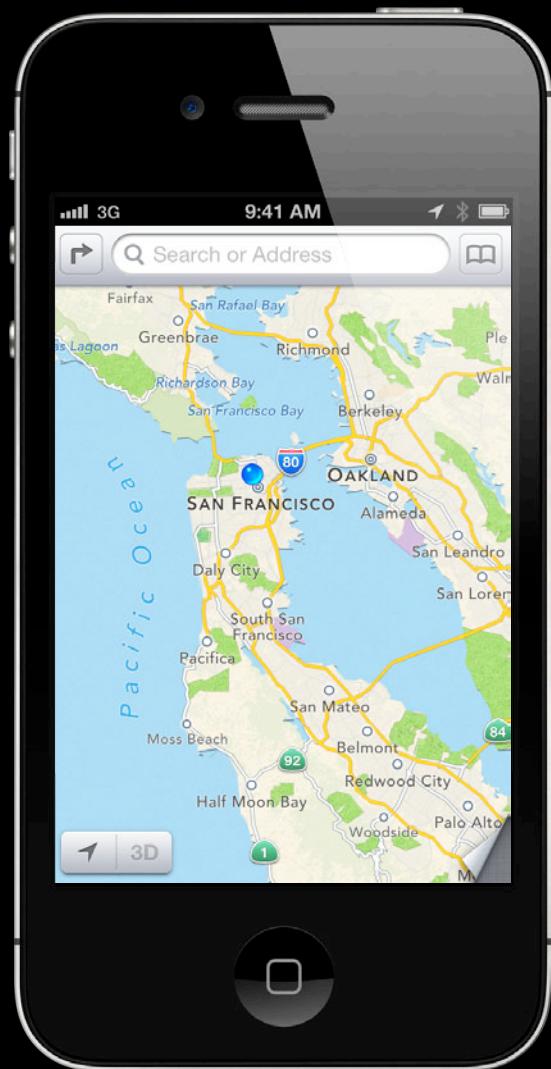


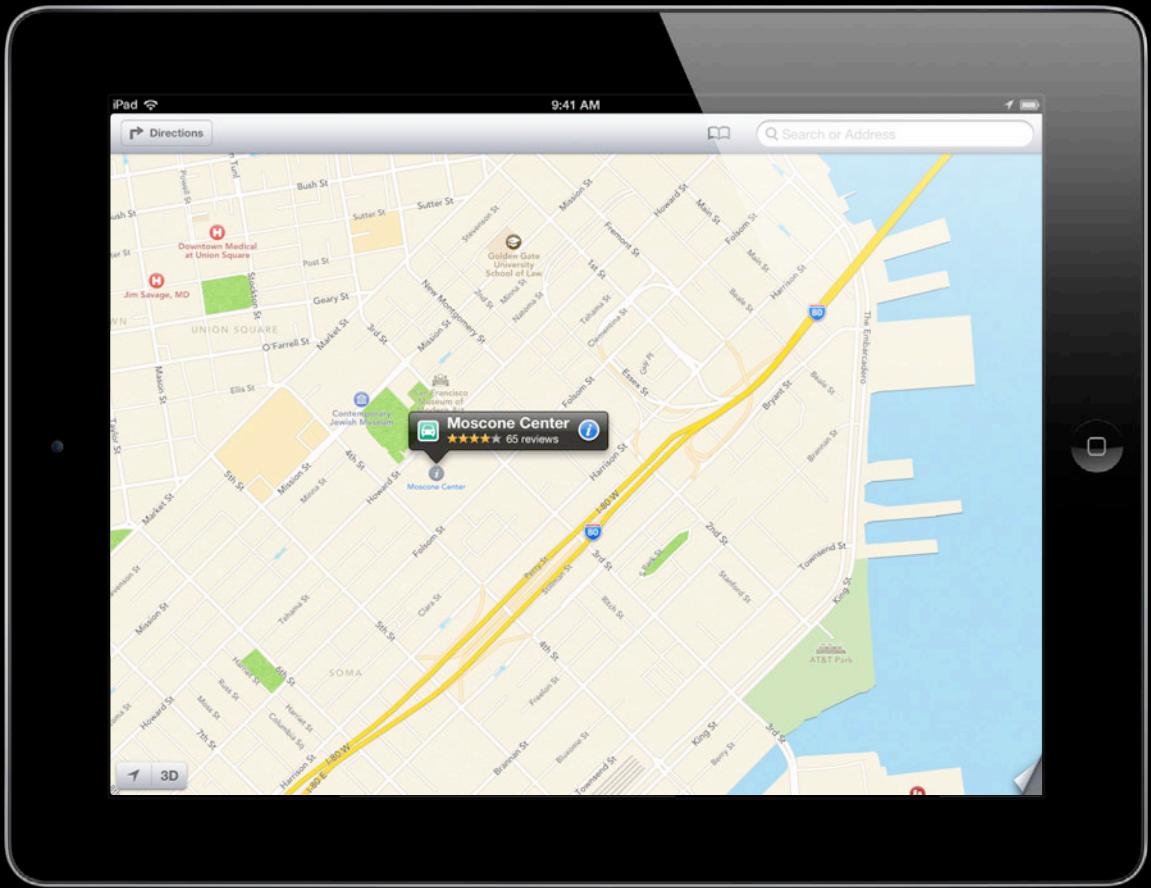
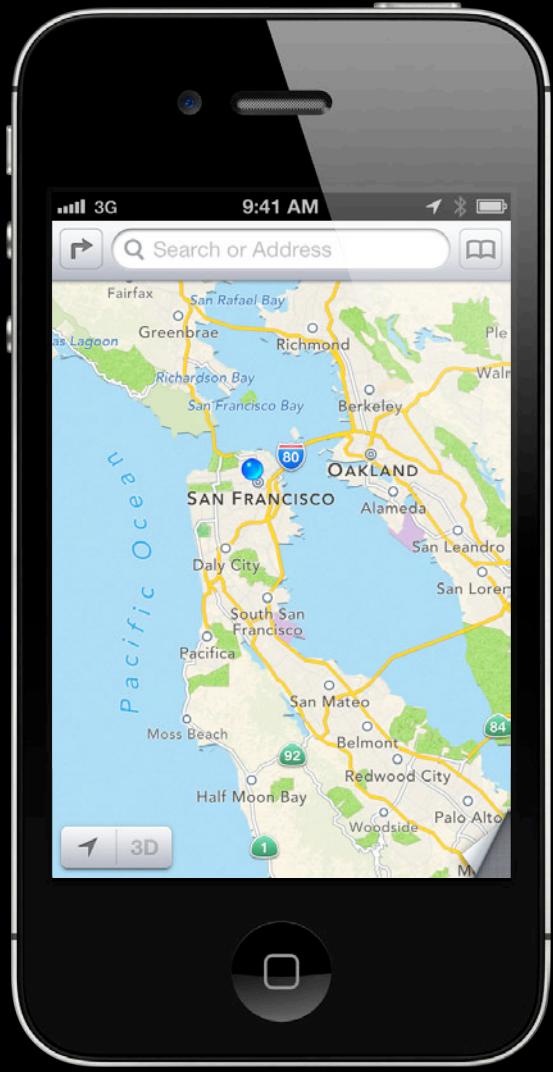
- The new look of Map Kit
- Launching Maps with MKMapItem
- Developing routing apps

The New Look of Map Kit

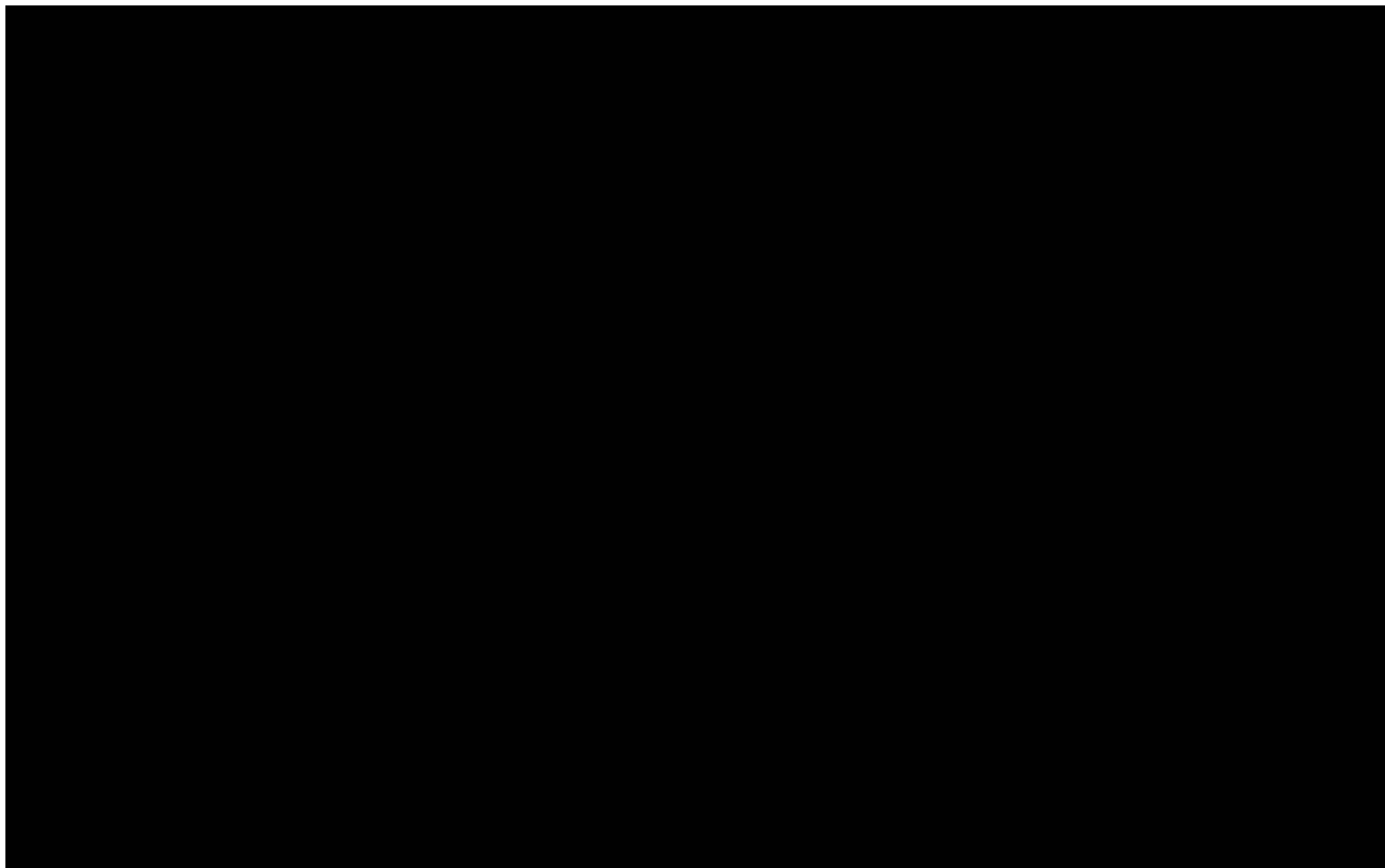
Powered by Apple Maps





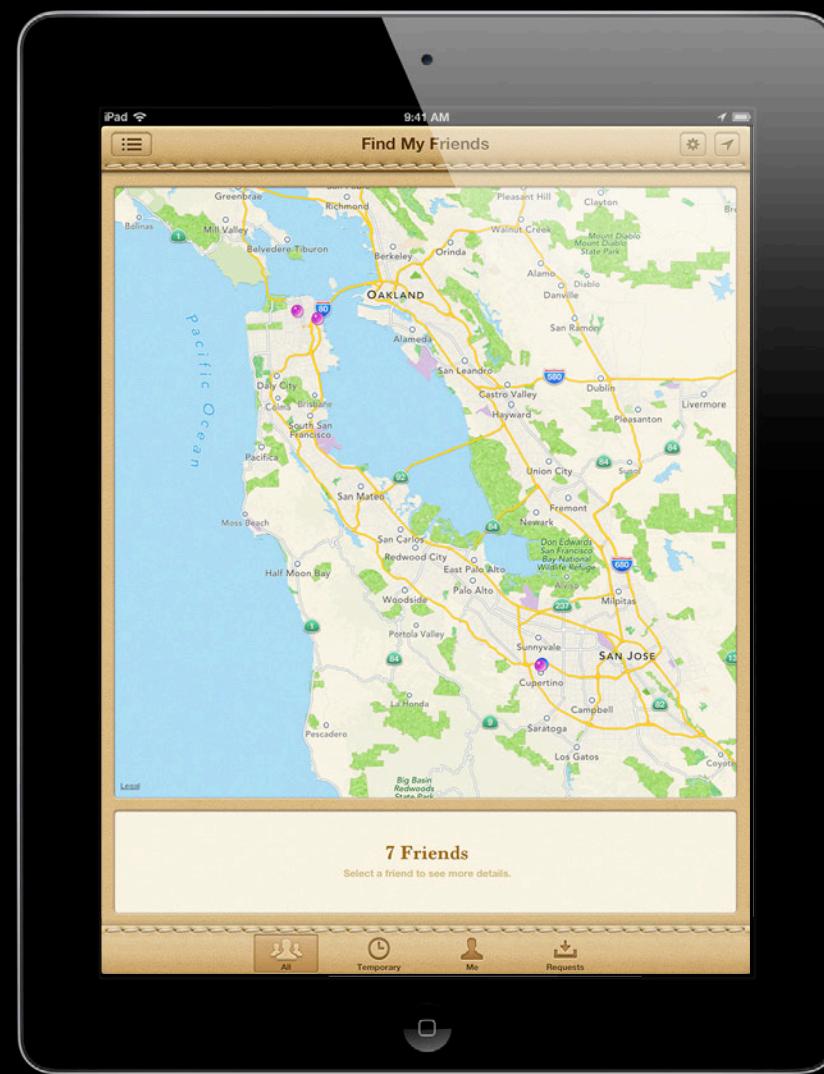


**What does it take to get
Apple Maps in my app?**



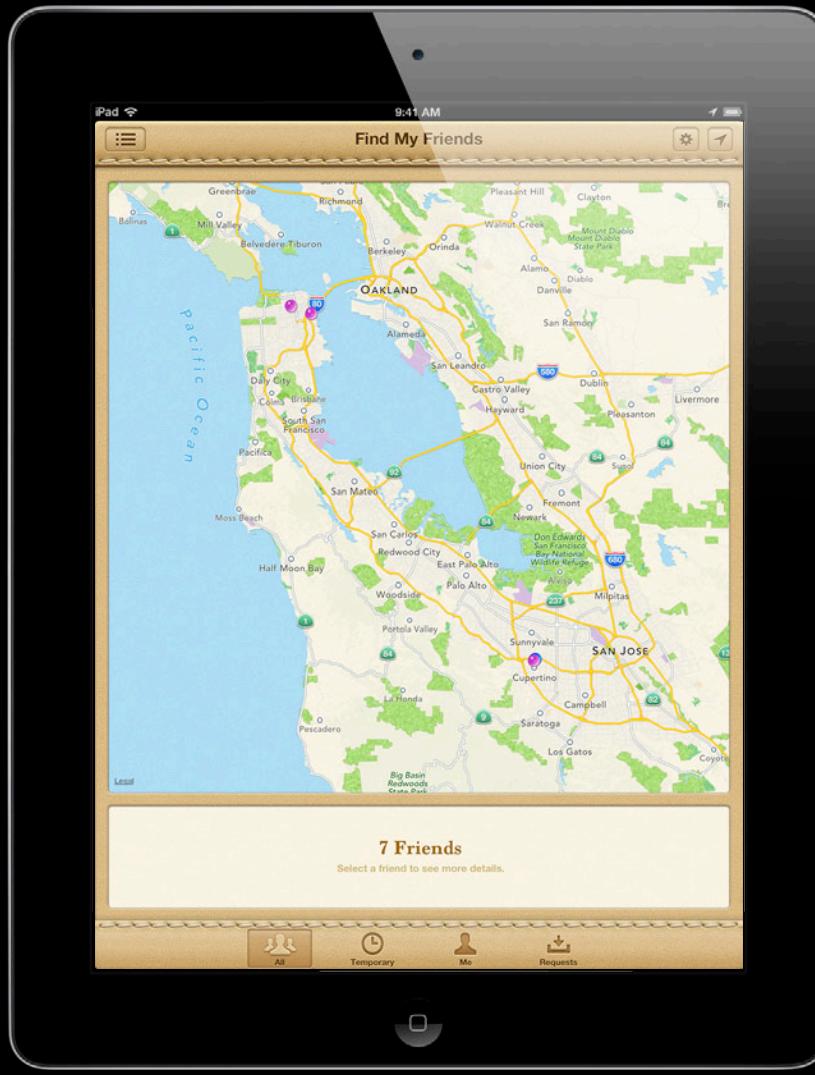
Absolutely nothing.

New Maps. **Same API.**



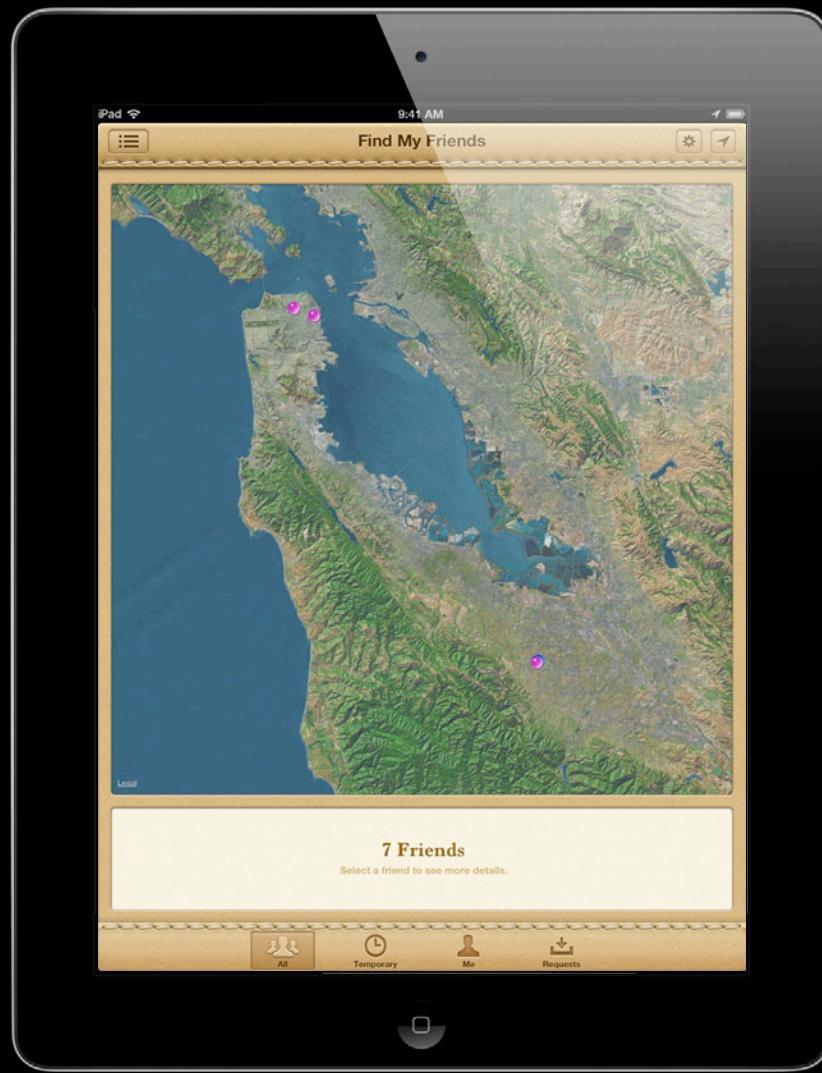
Map Modes

Standard



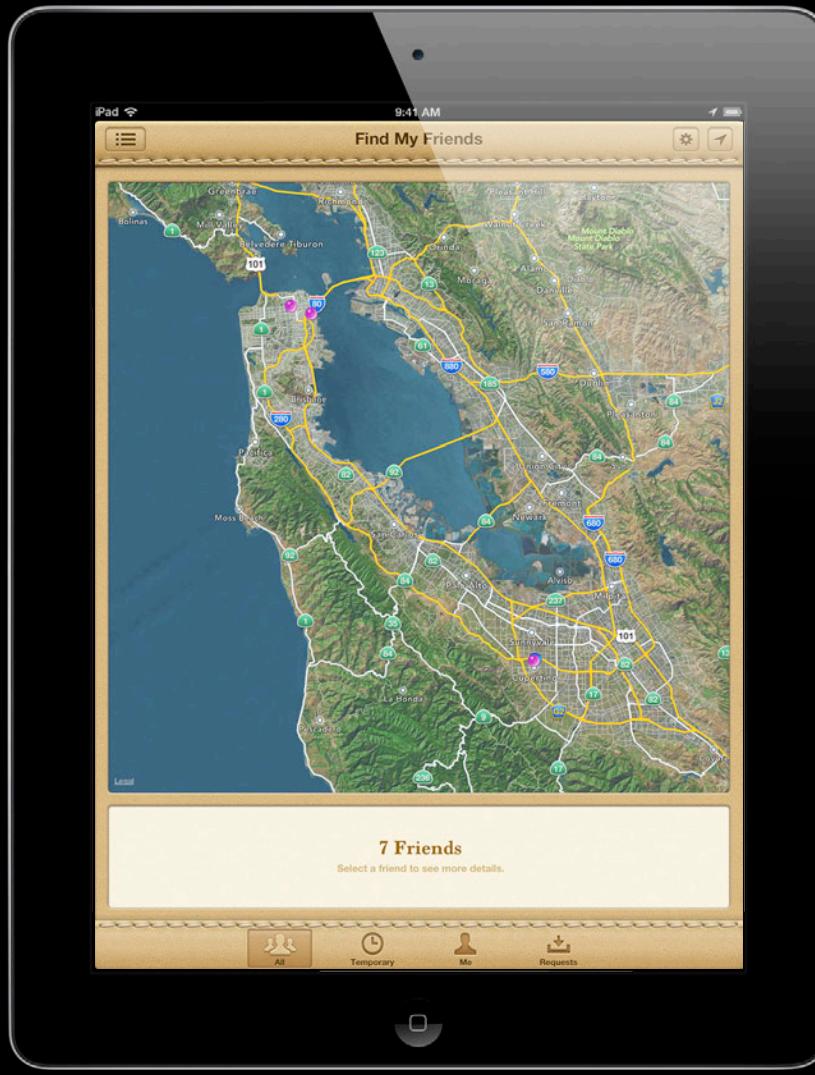
Map Modes

Satellite

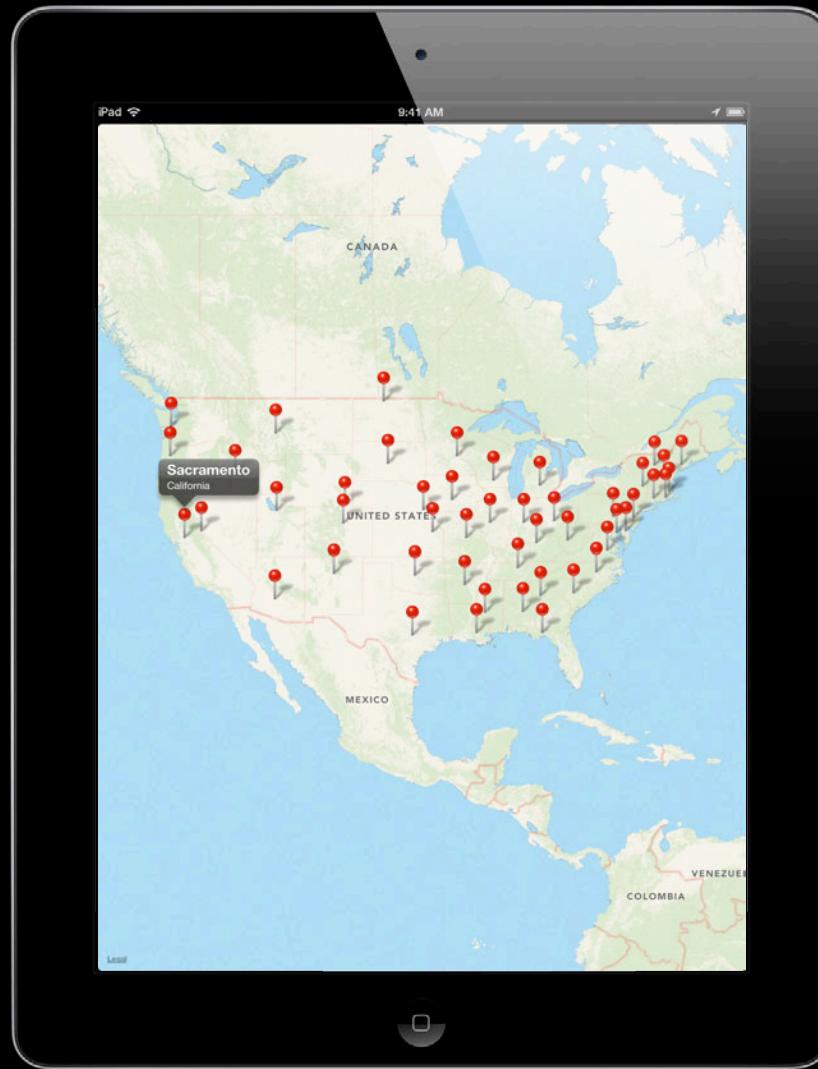


Map Modes

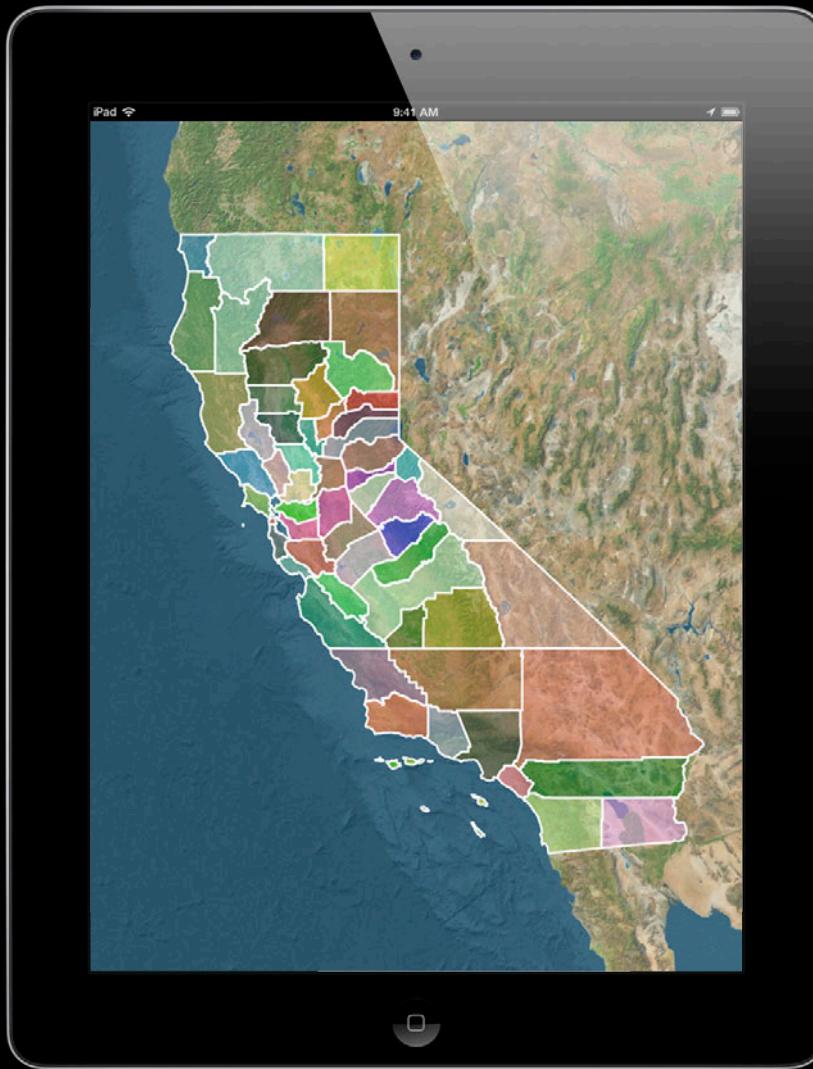
Hybrid



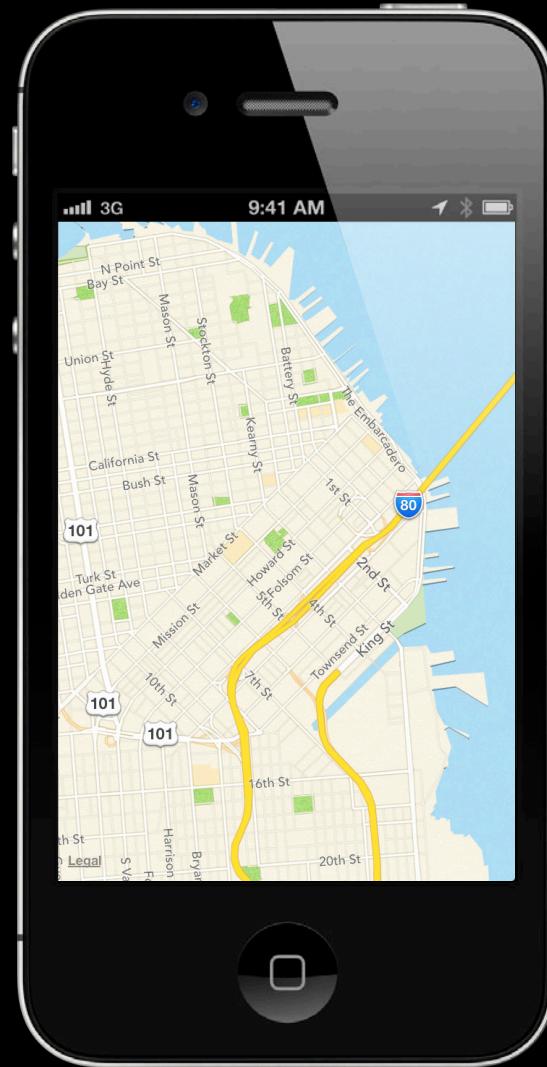
Annotations



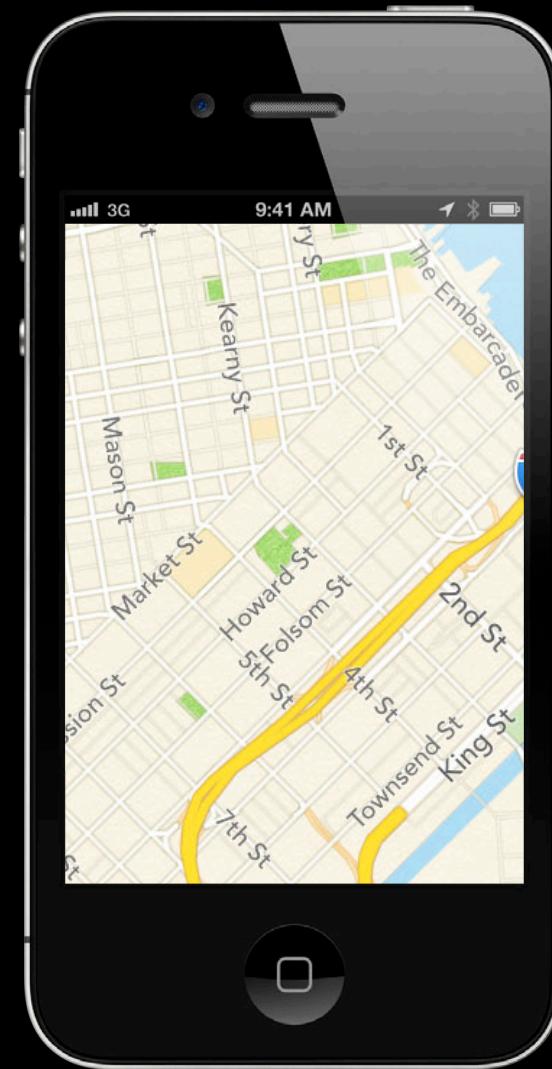
Overlays



Pinch to Zoom With a Static Map



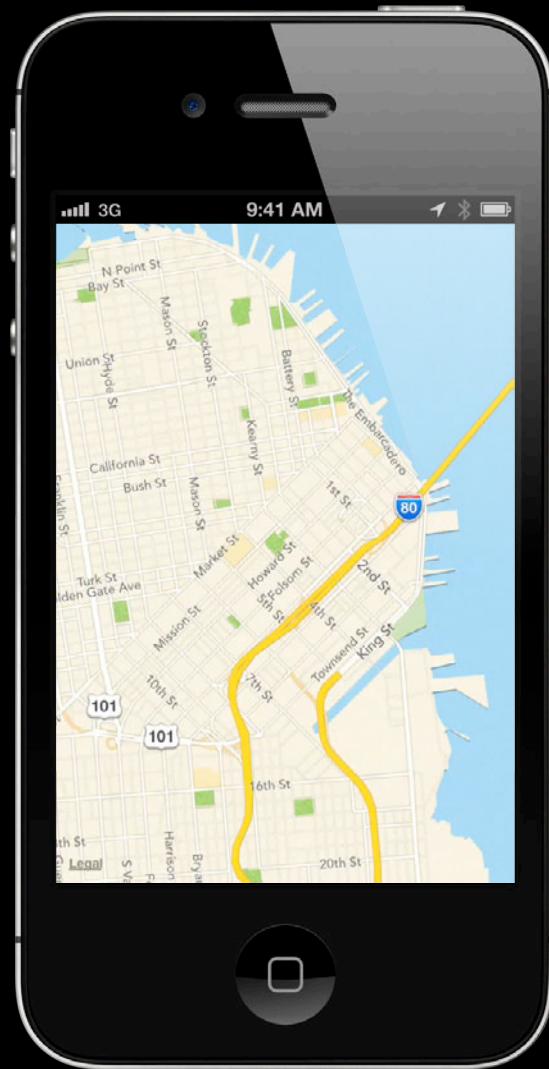
Pinch to Zoom With a Static Map

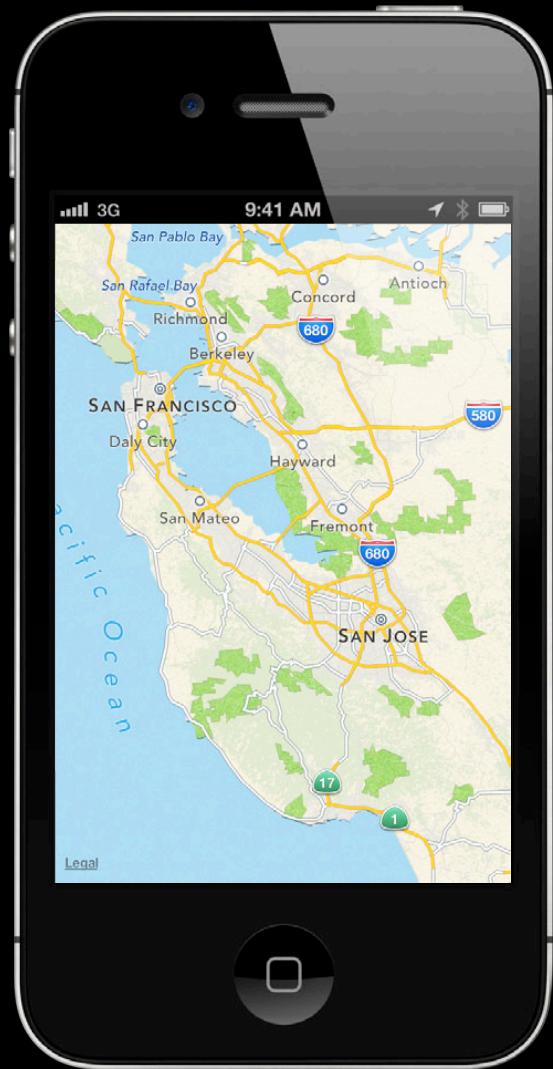


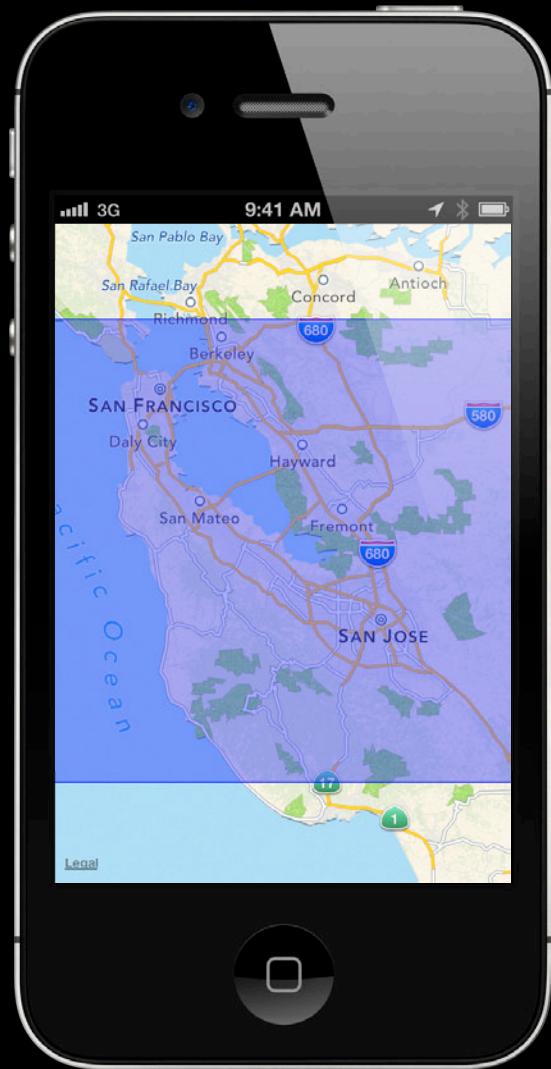
Pinch to Zoom With a Dynamic Map

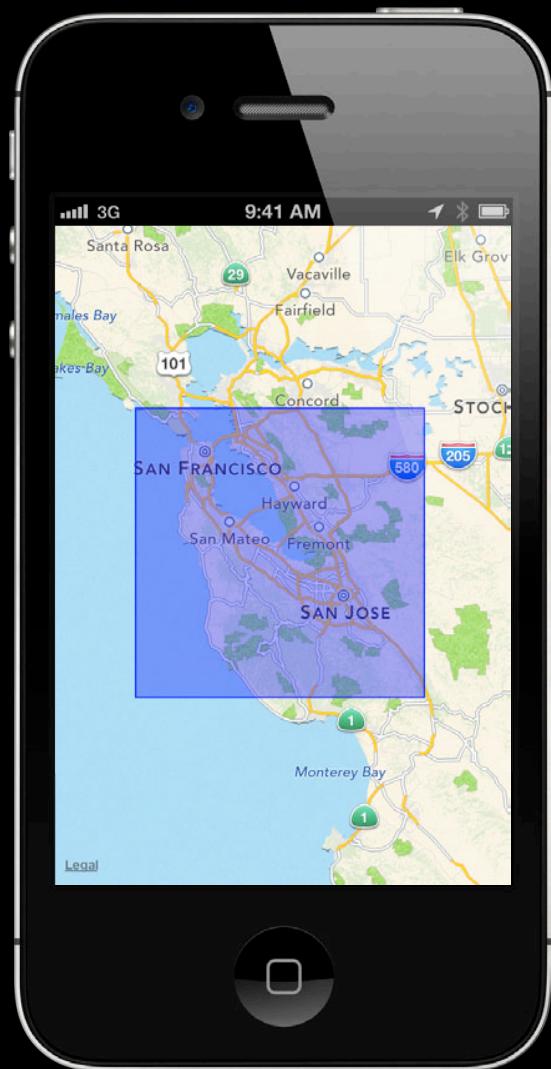


Pinch to Zoom With a Dynamic Map

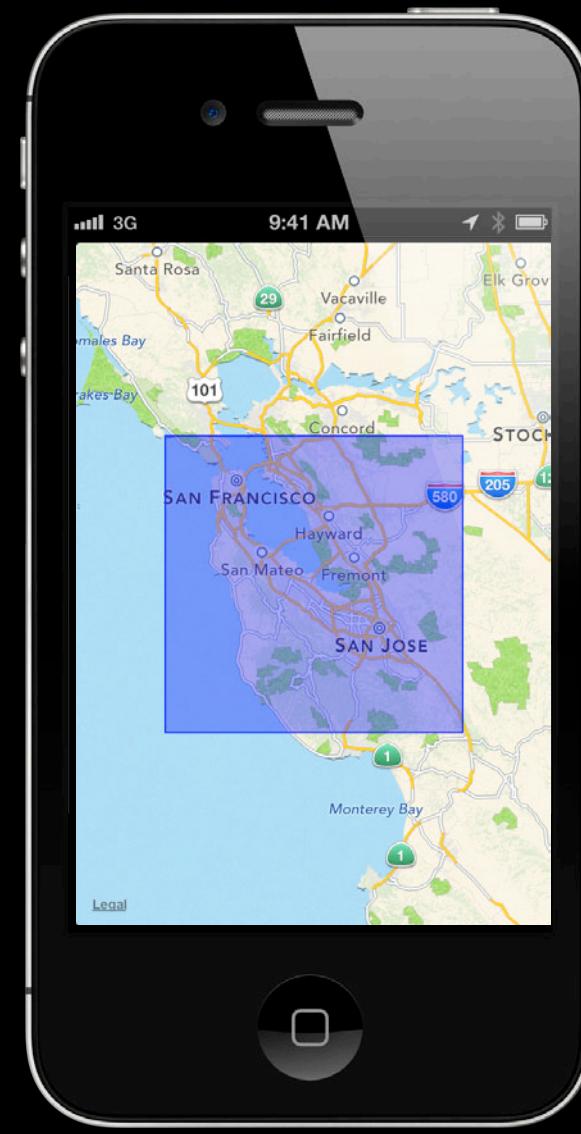






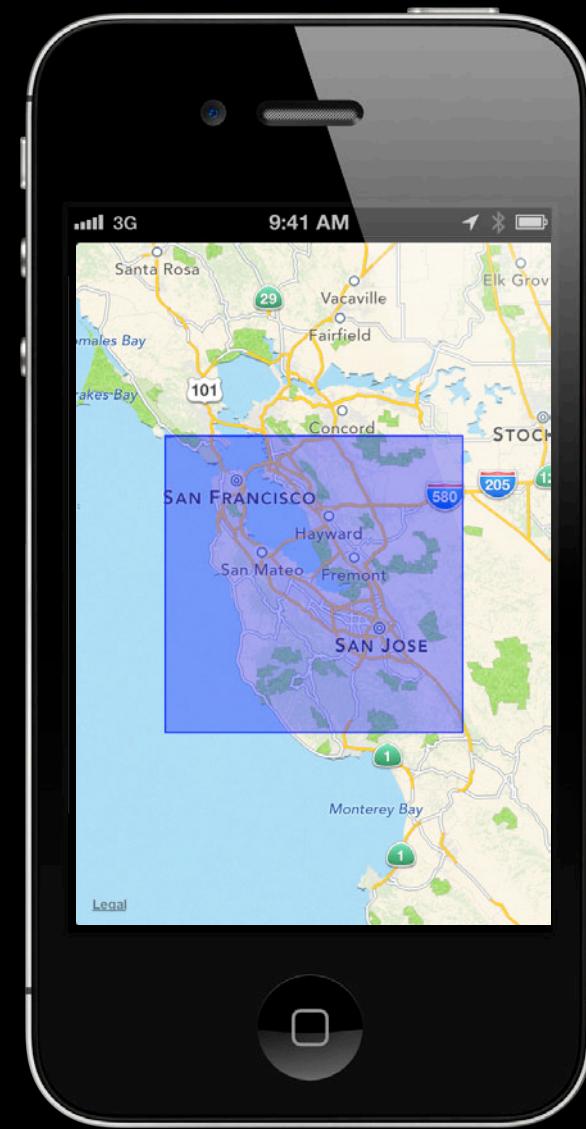


Setting the Map Region



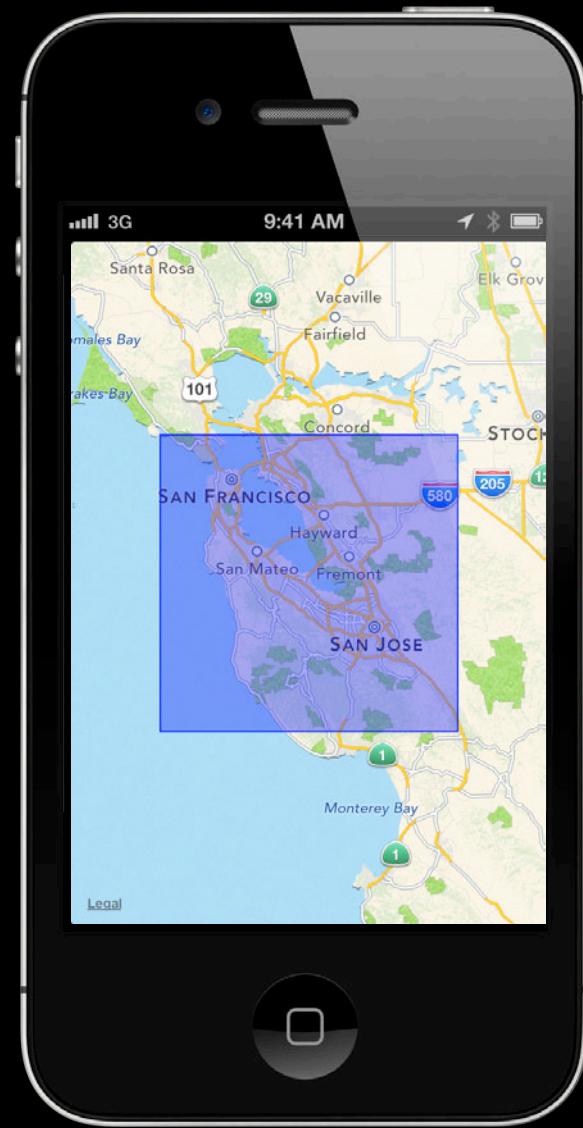
Setting the Map Region

- In iOS 5, region changes were adjusted to improve map display



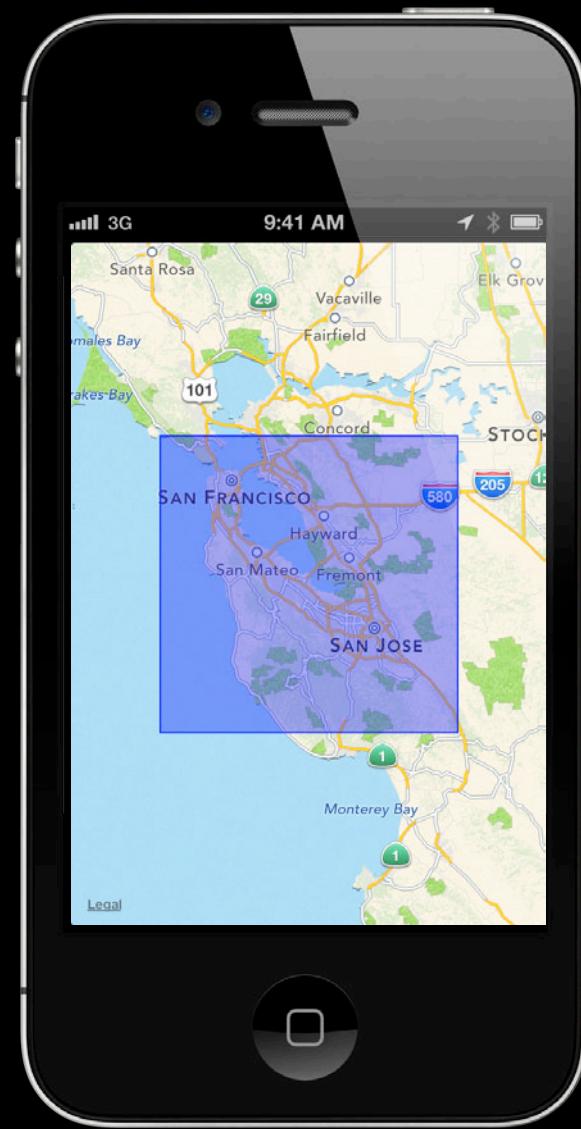
Setting the Map Region

- In iOS 5, region changes were adjusted to improve map display
 - setRegion:



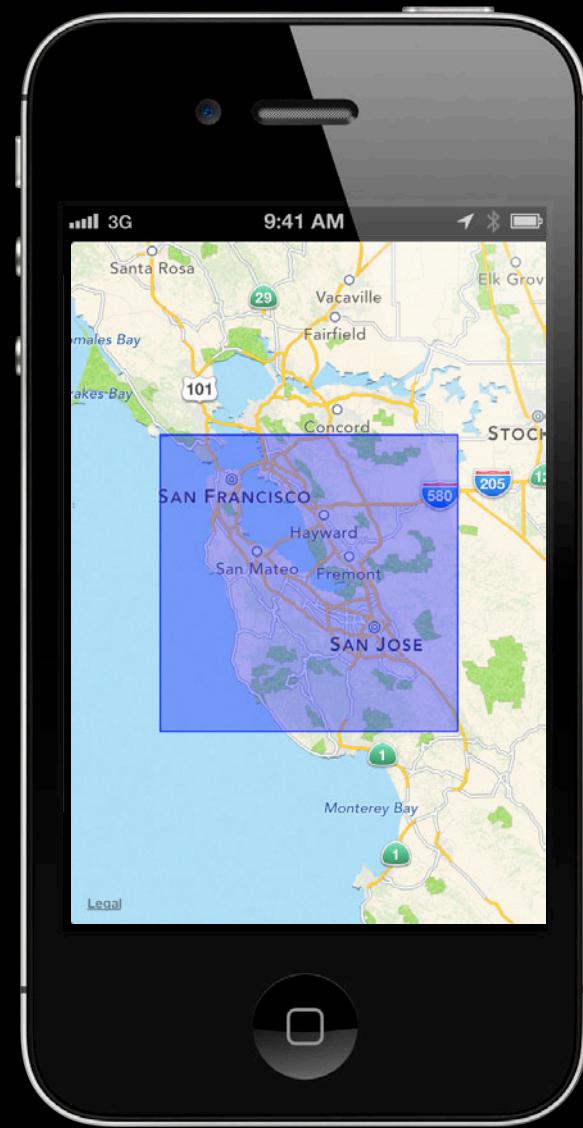
Setting the Map Region

- In iOS 5, region changes were adjusted to improve map display
 - setRegion:
 - setVisibleMapRect:



Setting the Map Region

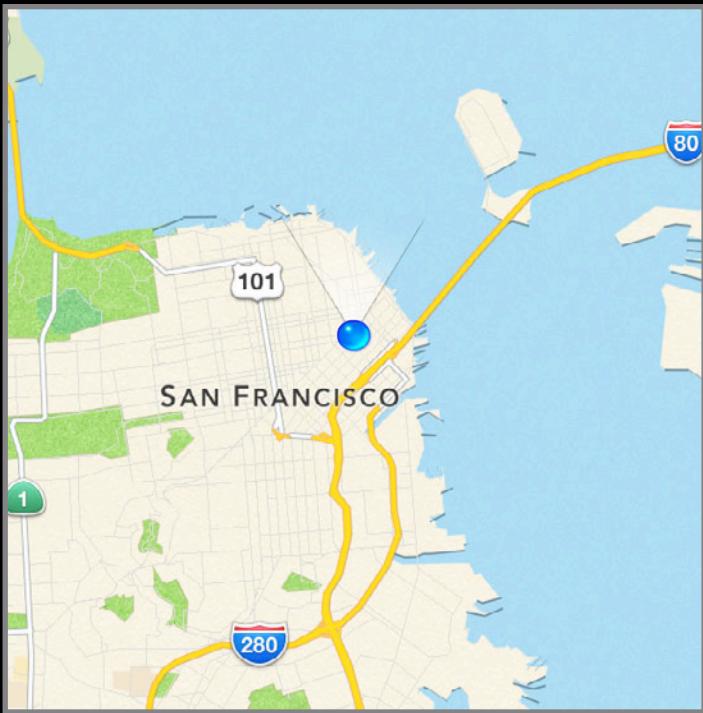
- In iOS 5, region changes were adjusted to improve map display
 - setRegion:
 - setVisibleMapRect:
- No longer necessary in iOS 6



Heading Mode

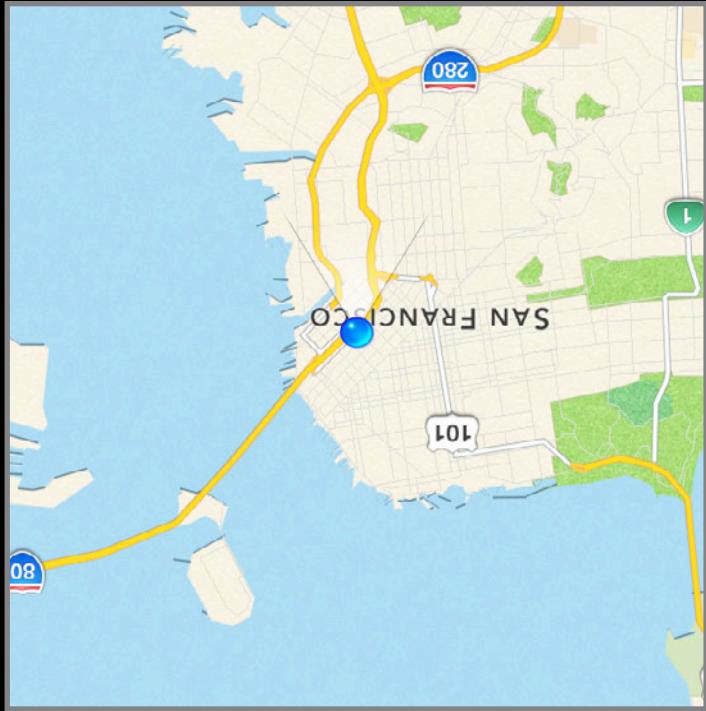
Heading Mode

Static Map (iOS 5)



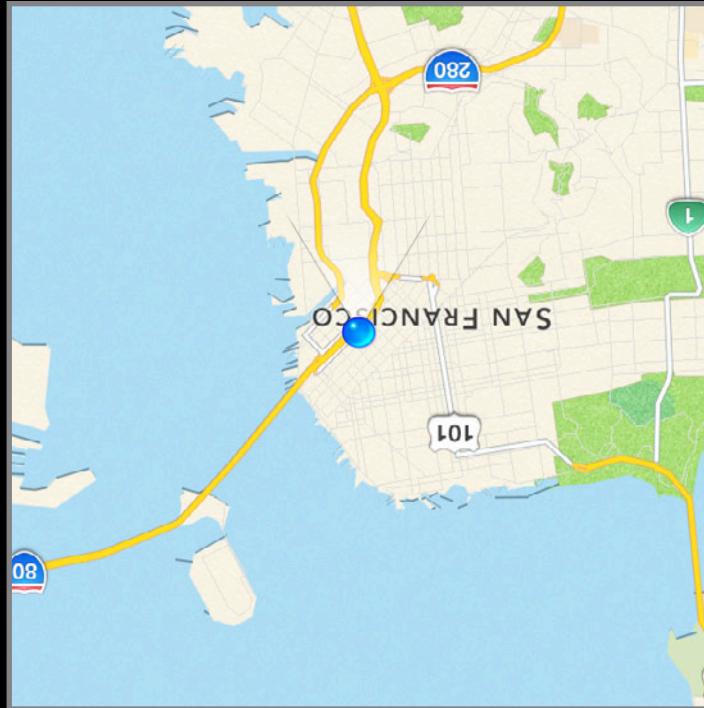
Heading Mode

Static Map (iOS 5)

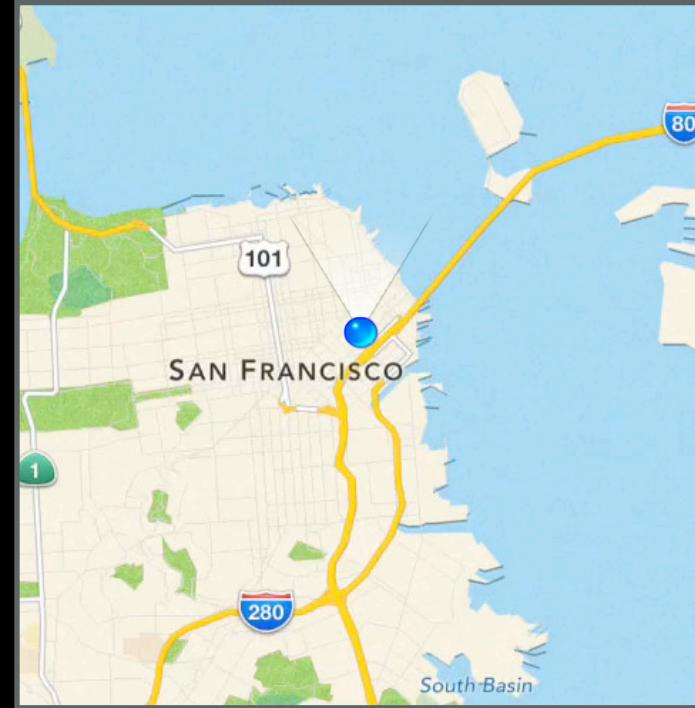


Heading Mode

Static Map (iOS 5)

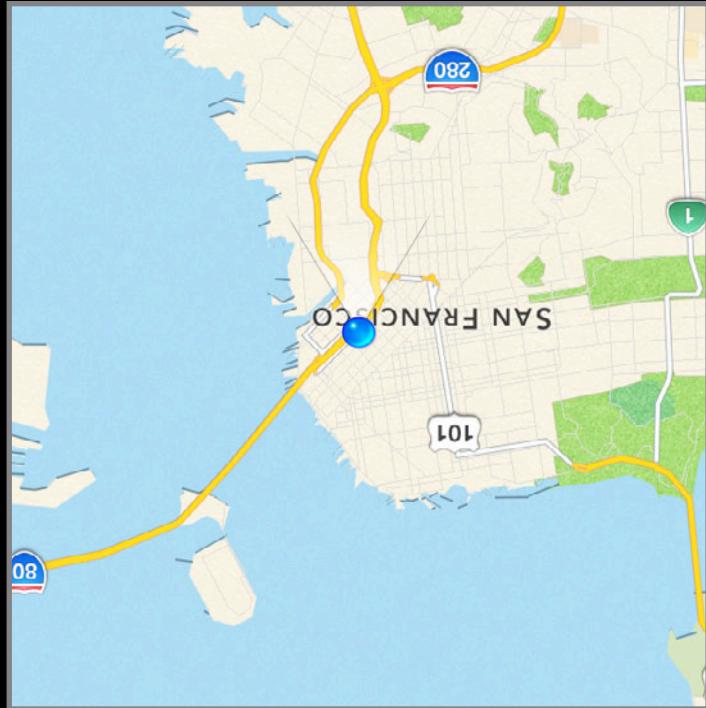


Dynamic Map (iOS 6)

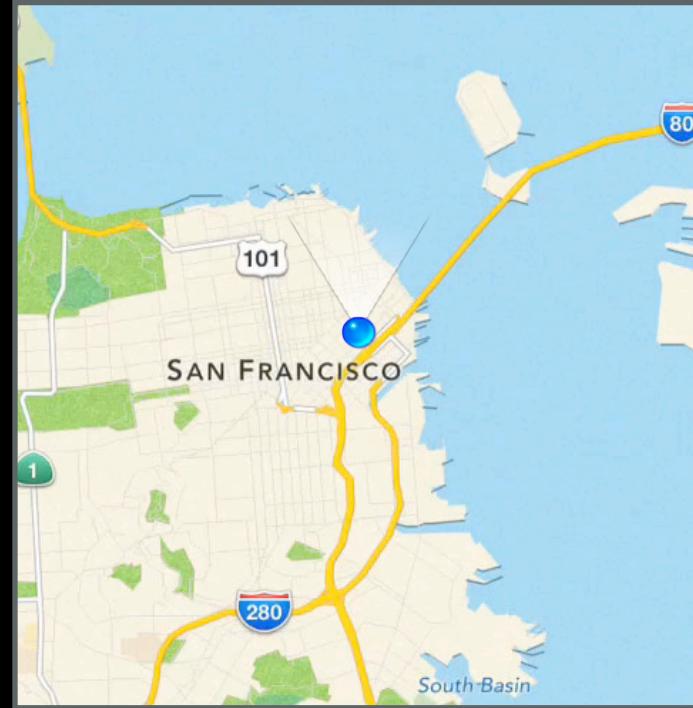


Heading Mode

Static Map (iOS 5)



Dynamic Map (iOS 6)



The New Look of Map Kit



The New Look of Map Kit



- Map display APIs are the same

The New Look of Map Kit



- Map display APIs are the same
- Improved map rendering

The New Look of Map Kit



- Map display APIs are the same
- Improved map rendering
- Re-link with iOS 6 to change regions with precision

The New Look of Map Kit



- Map display APIs are the same
- Improved map rendering
- Re-link with iOS 6 to change regions with precision
- Test your apps!

Launching Maps with MKMapItem

A new API in iOS 6

Why Launch Maps?

Getting directions



Why Launch Maps?

Drop a Pin



Contacts/Bookmarks



Explore



Introducing MKMapItem



Introducing MKMapItem



- Objective-C API

Introducing MKMapItem



- Objective-C API
- Open Maps with one or more pins

Introducing MKMapItem



- Objective-C API
- Open Maps with one or more pins
- Jump directly into directions

Introducing MKMapItem

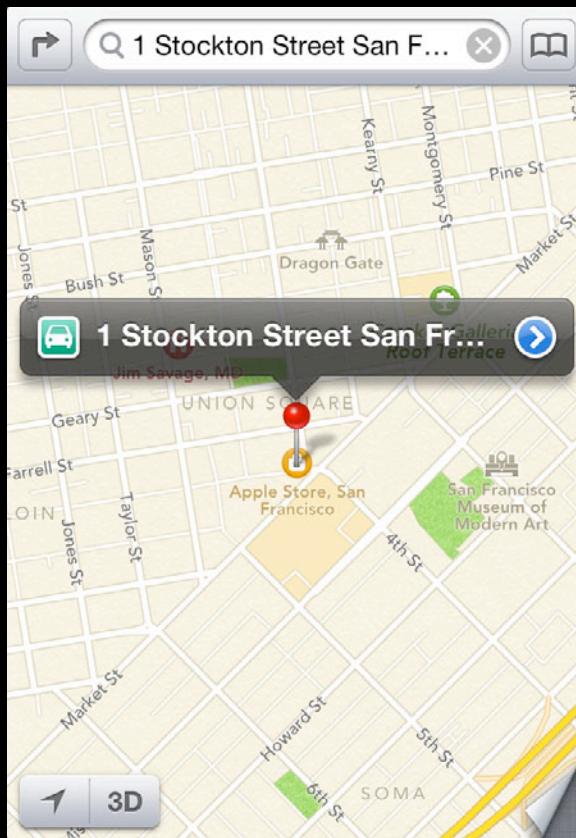


- Objective-C API
- Open Maps with one or more pins
- Jump directly into directions
- Customize the map display

Opening an Item in Maps



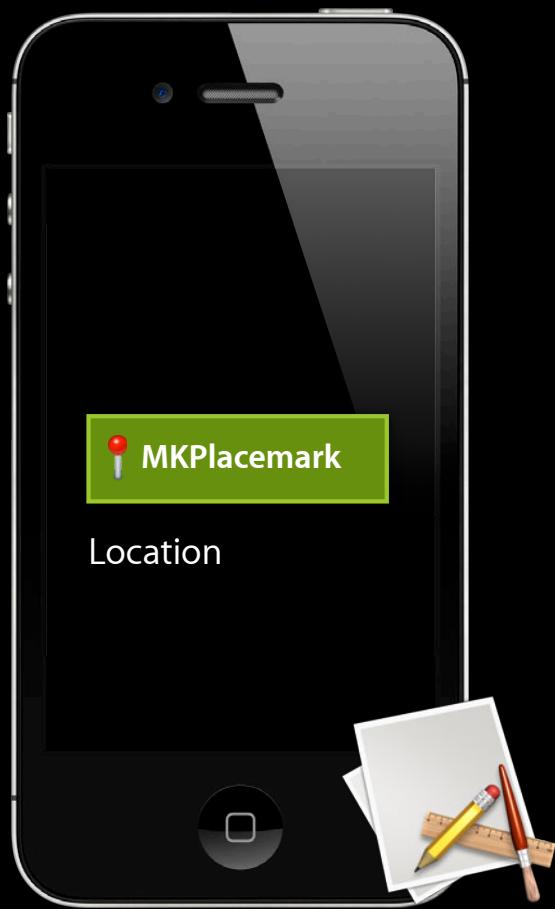
Opening an Item in Maps



Opening an Item in Maps



Opening an Item in Maps



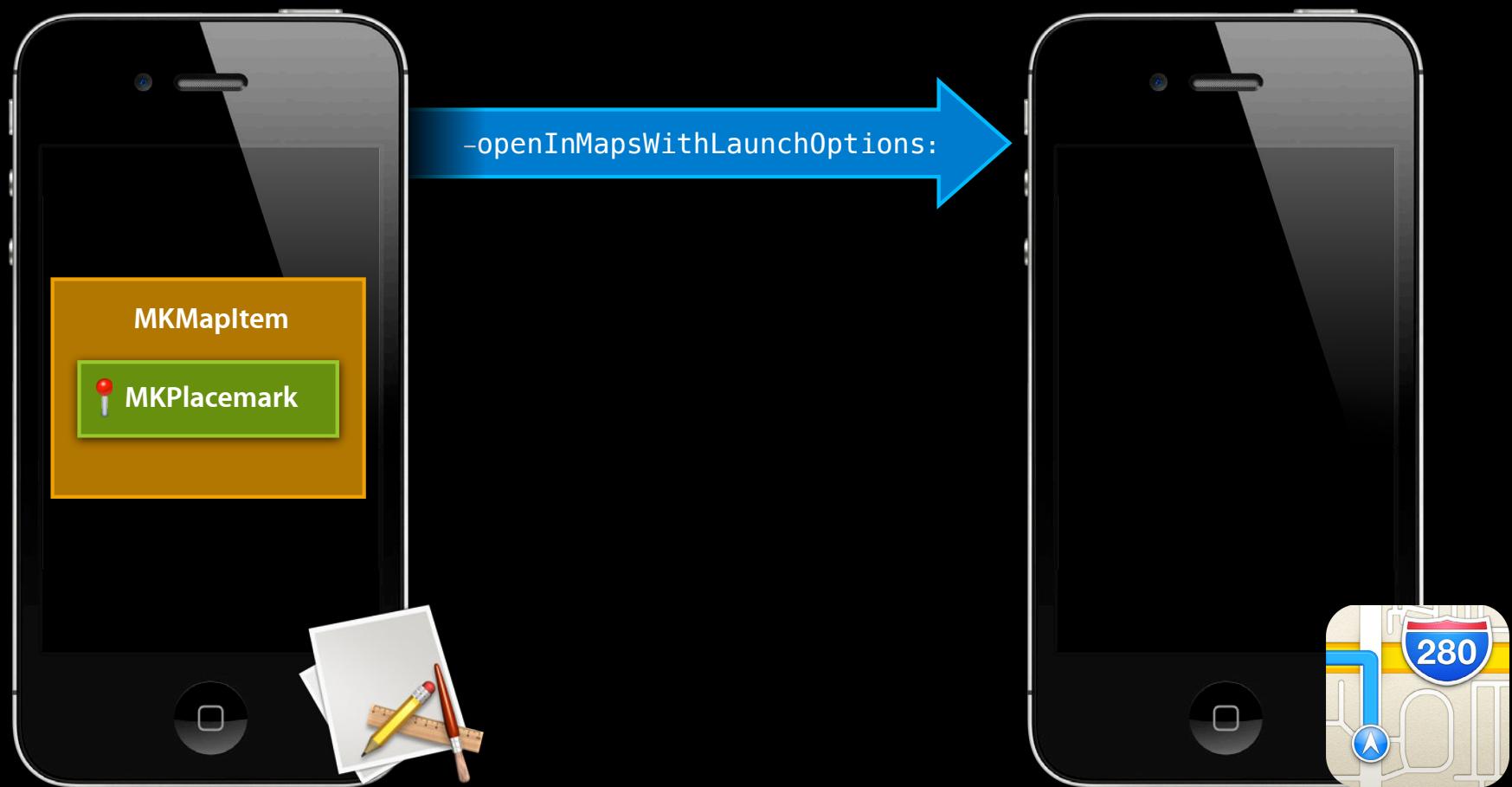
Opening an Item in Maps



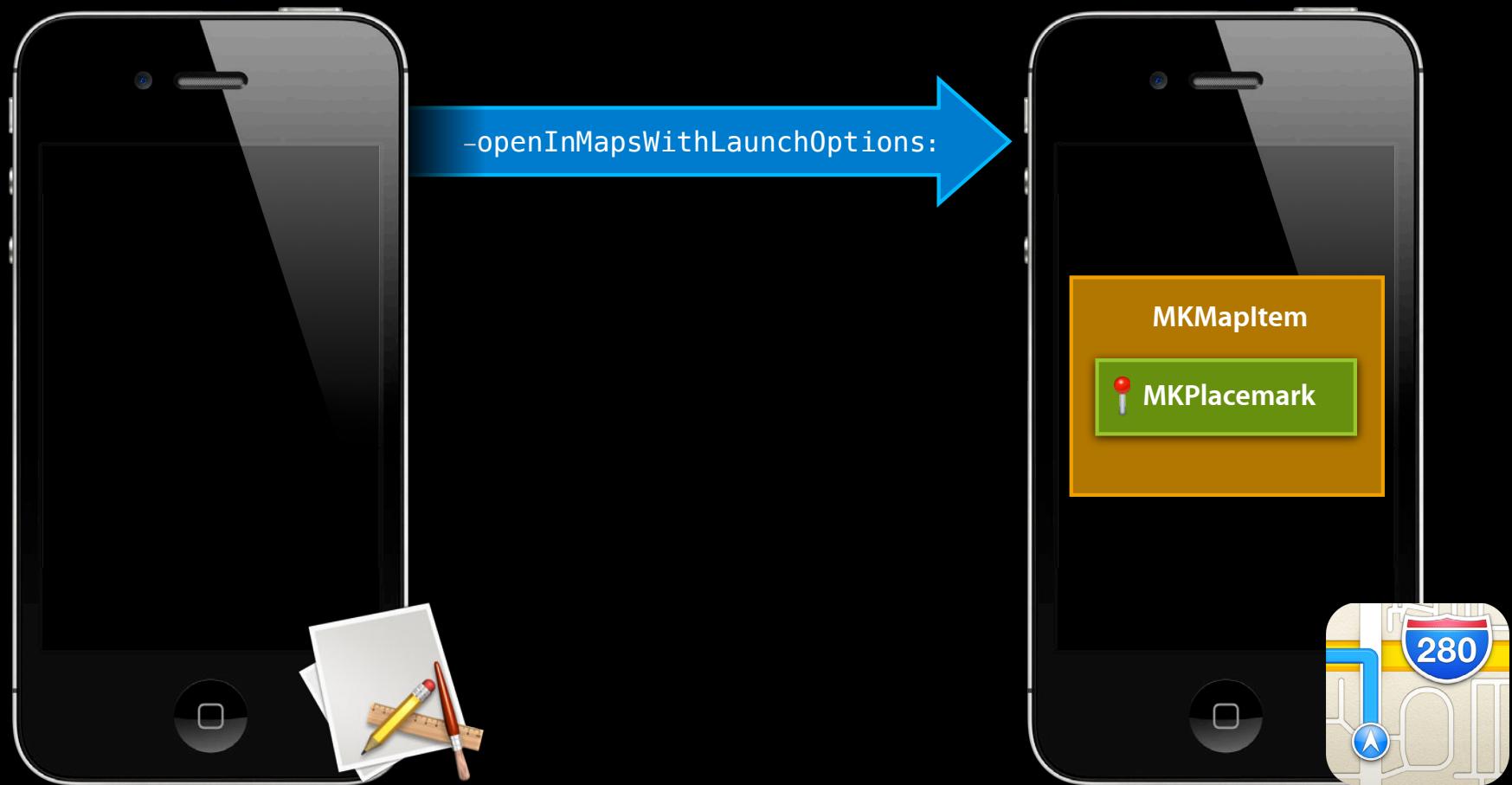
Opening an Item in Maps



Opening an Item in Maps



Opening an Item in Maps



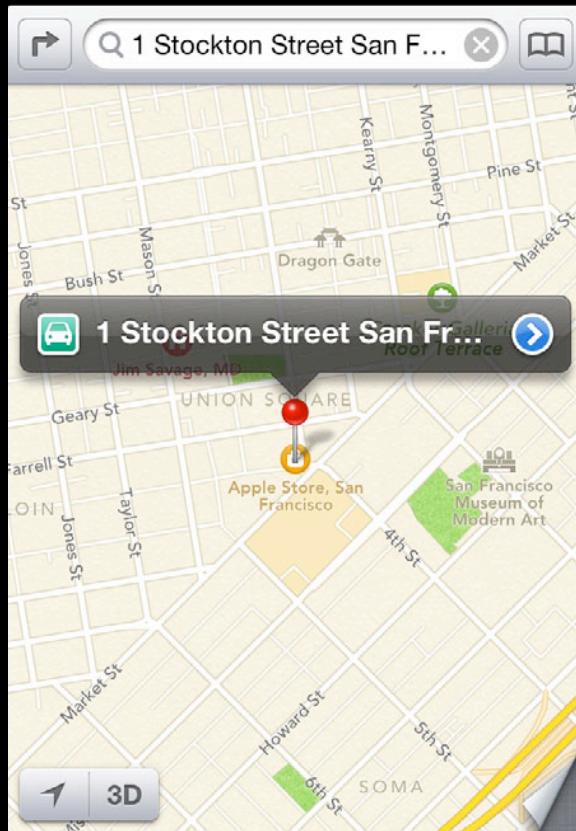
Opening an Item in Maps

```
MKMapItem *sfStore = [ [MKMapItem alloc] initWithPlacemark:placemark];
```

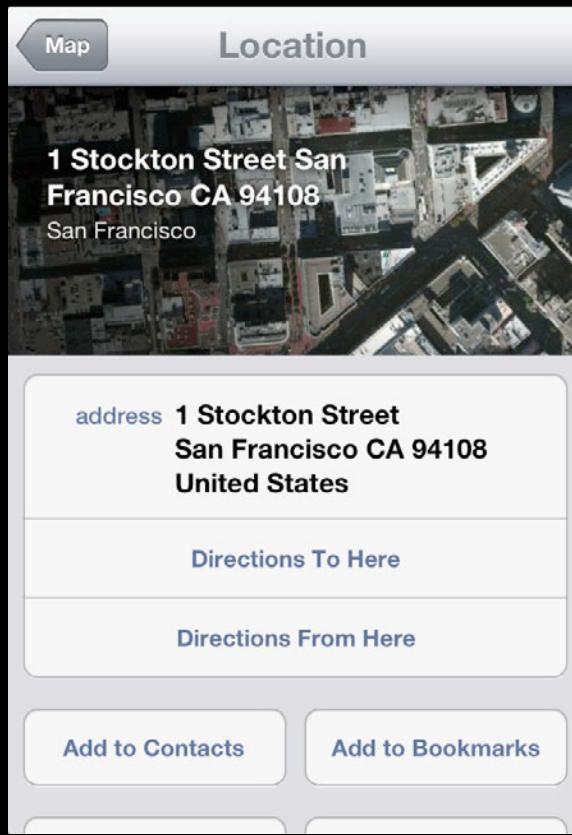
Opening an Item in Maps

```
MKMapItem *sfStore = [ [MKMapItem alloc] initWithPlacemark:placemark];  
[sfStore openInMapsWithLaunchOptions:nil];
```

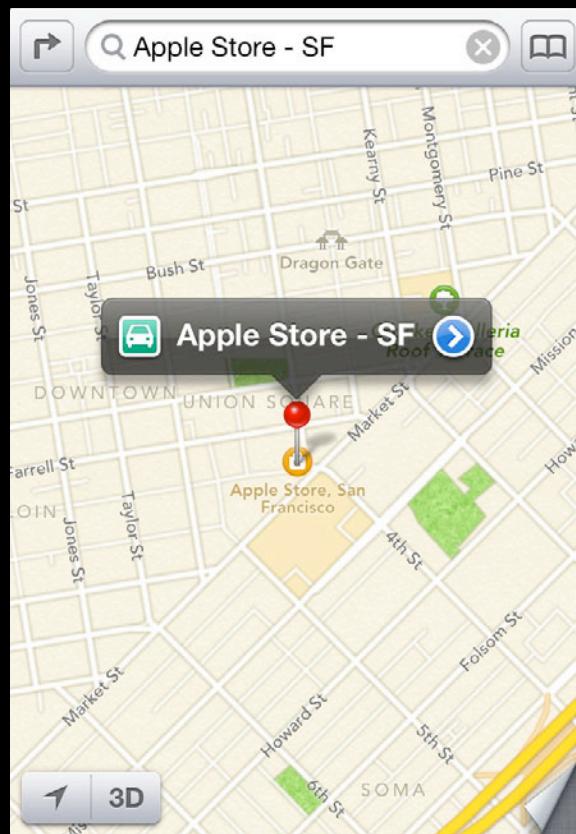
Opening an Item in Maps



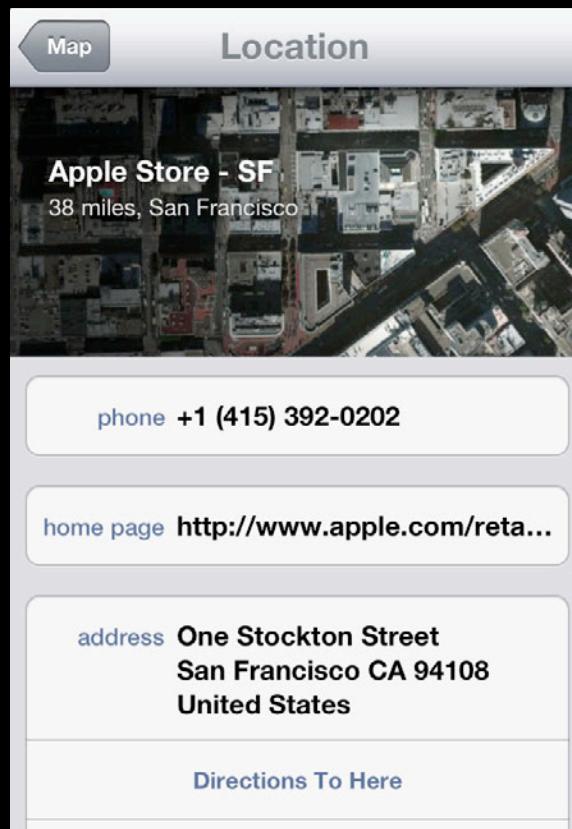
Opening an Item in Maps



Adding Item Details



Adding Item Details



Adding Item Details



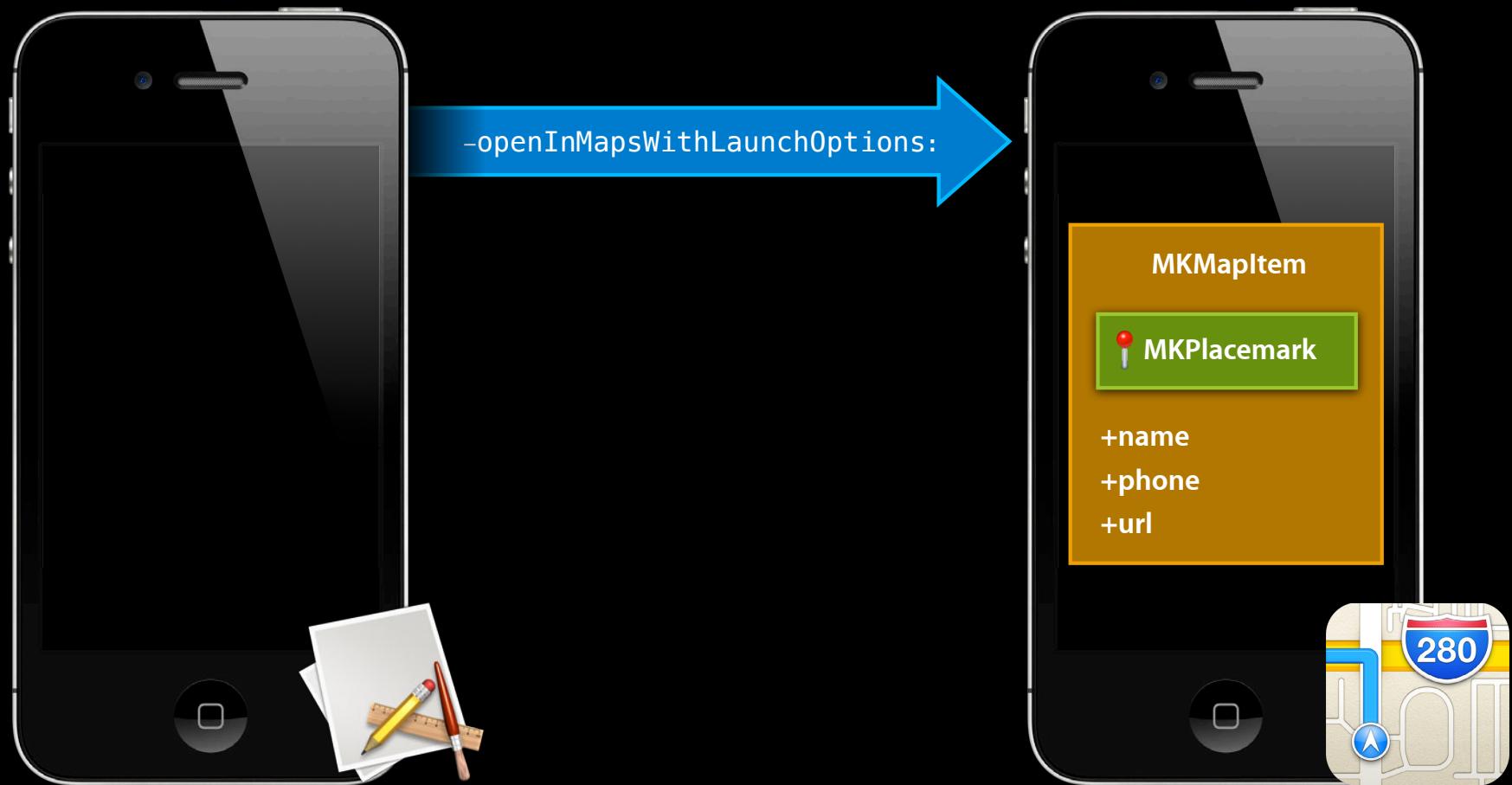
Adding Item Details



Adding Item Details



Adding Item Details



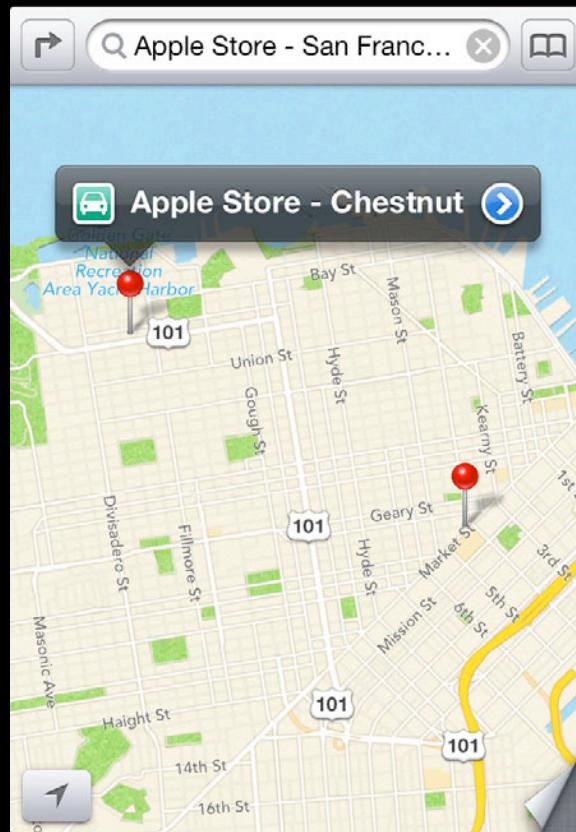
Adding Item Details

```
MKMapItem *sfStore = [ [MKMapItem alloc] initWithPlacemark:placemark];  
  
sfStore.name = @"Apple Store - San Francisco";  
sfStore.phoneNumber = @"+14153920202";  
sfStore.url = [NSURL URLWithString:@"http://www.apple.com/retail/"];  
  
[sfStore openInMapsWithLaunchOptions:nil];
```

Opening Multiple Map Items



Opening Multiple Map Items



Opening Multiple Items in Maps



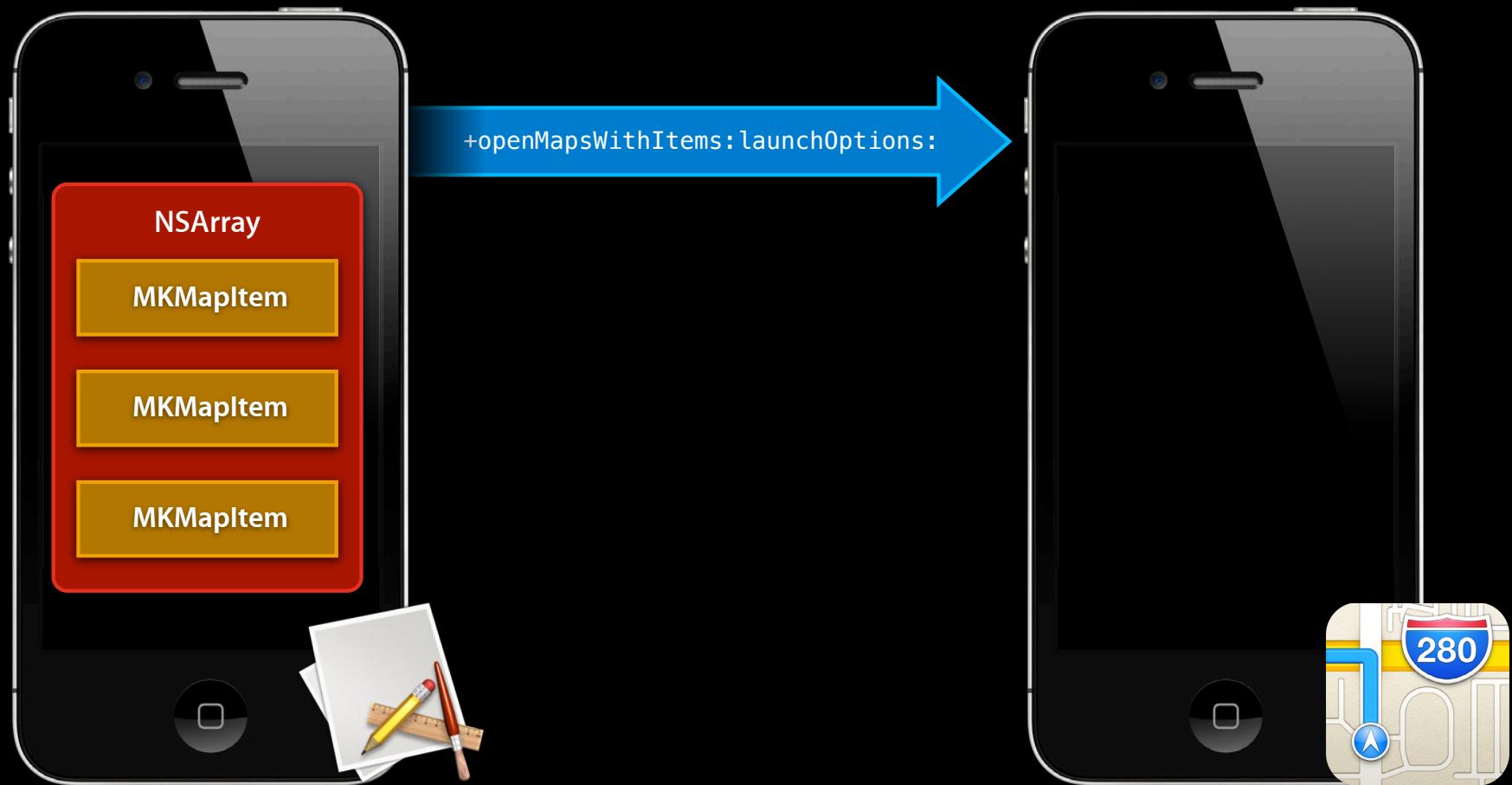
Opening Multiple Items in Maps



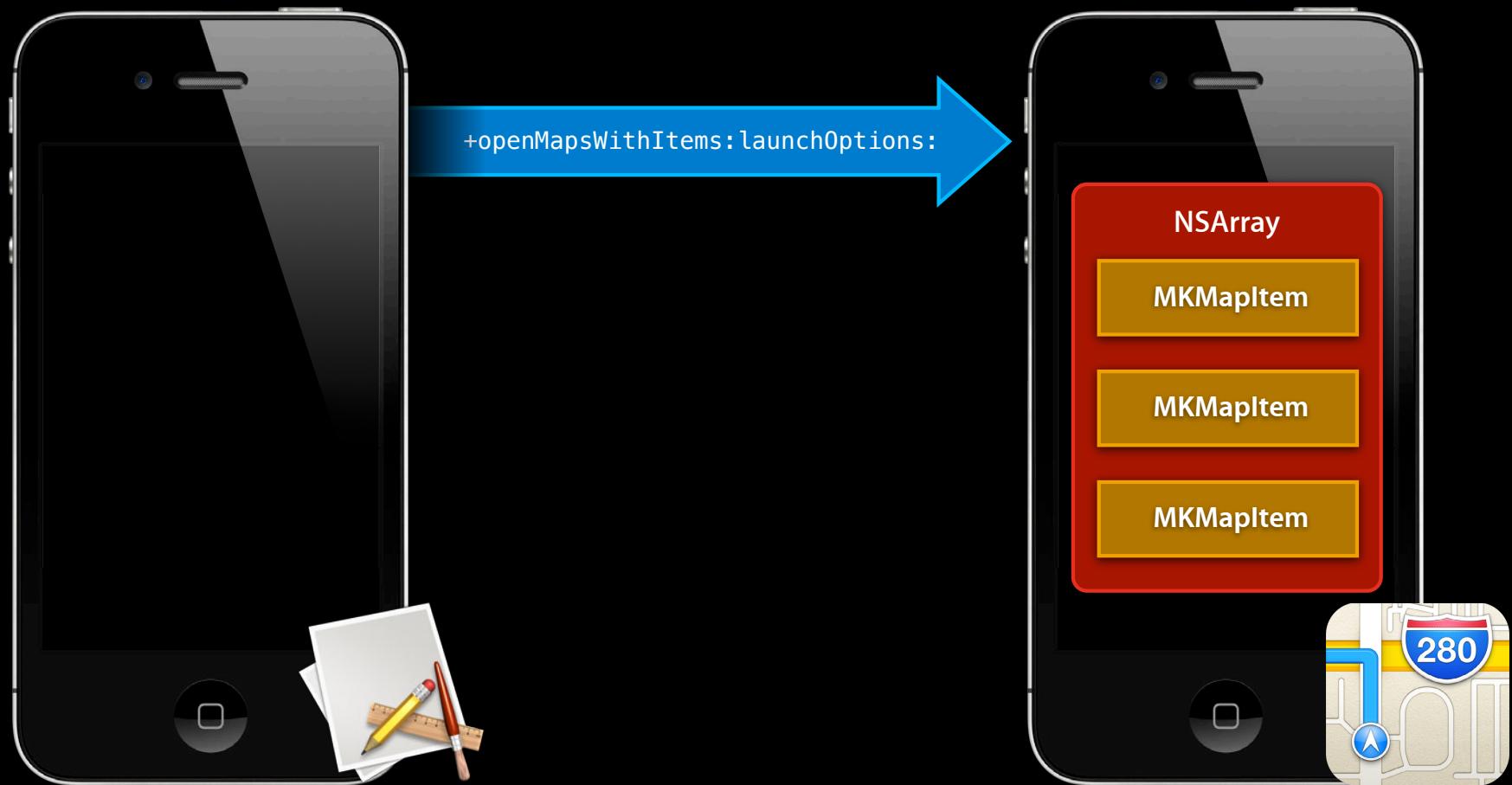
Opening Multiple Items in Maps



Opening Multiple Items in Maps



Opening Multiple Items in Maps



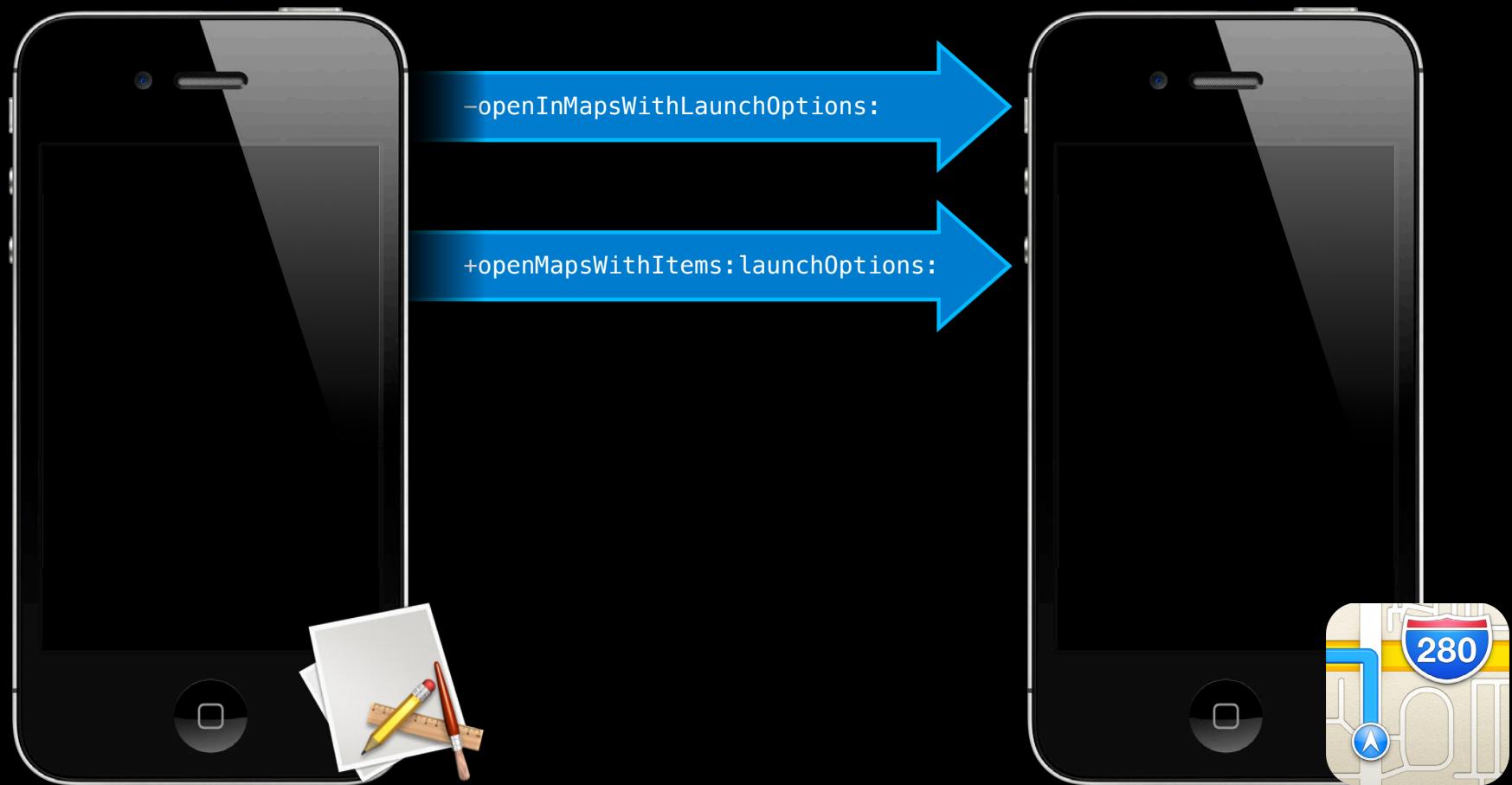
Opening Multiple Map Items

```
NSArray *storeItems = @[ sfStore, chestnutStore, bayStreetStore ];
```

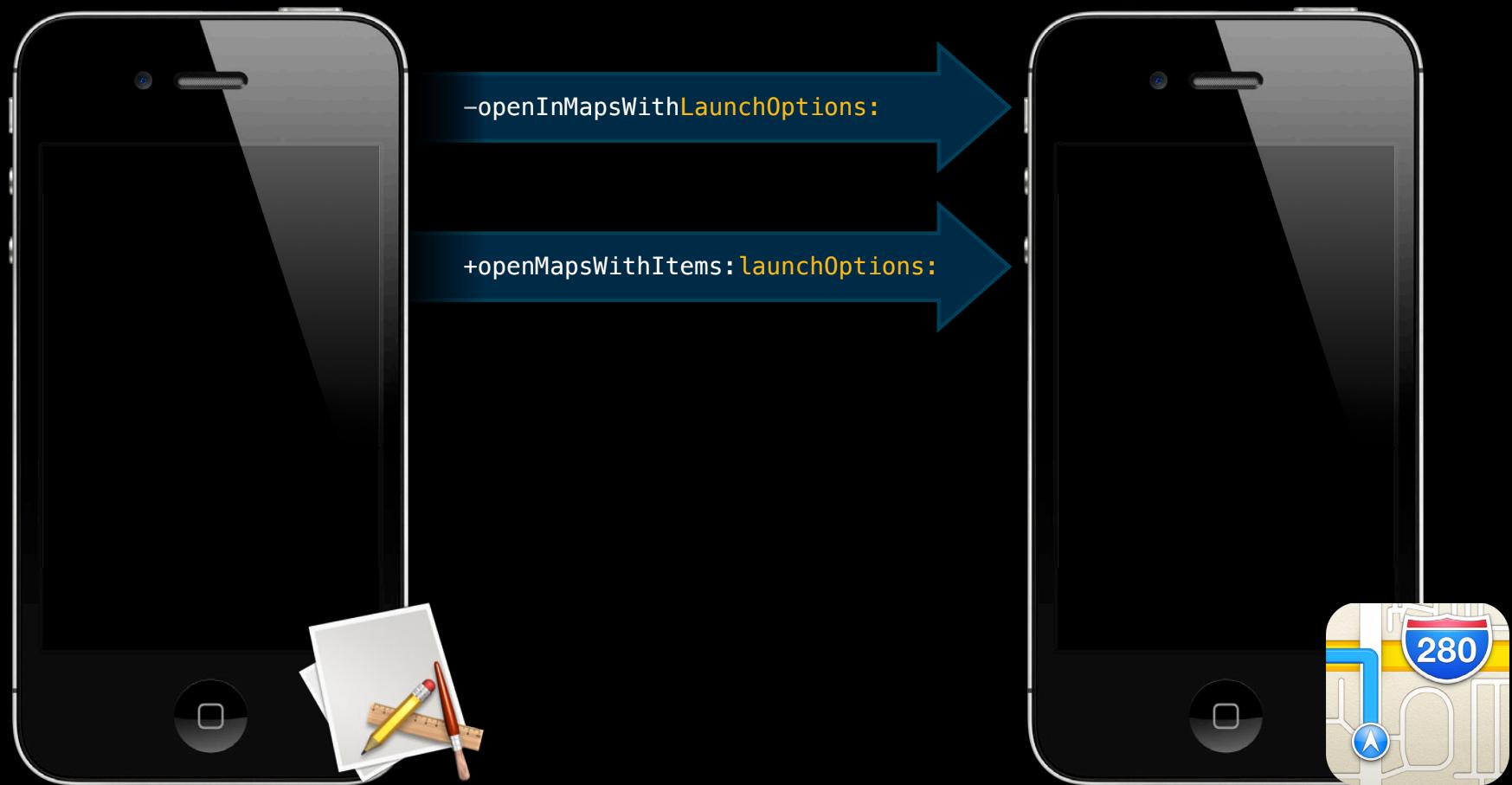
Opening Multiple Map Items

```
NSArray *storeItems = @[ sfStore, chestnutStore, bayStreetStore ];  
[MKMapItem openMapsWithItems:storeItems launchOptions:nil];
```

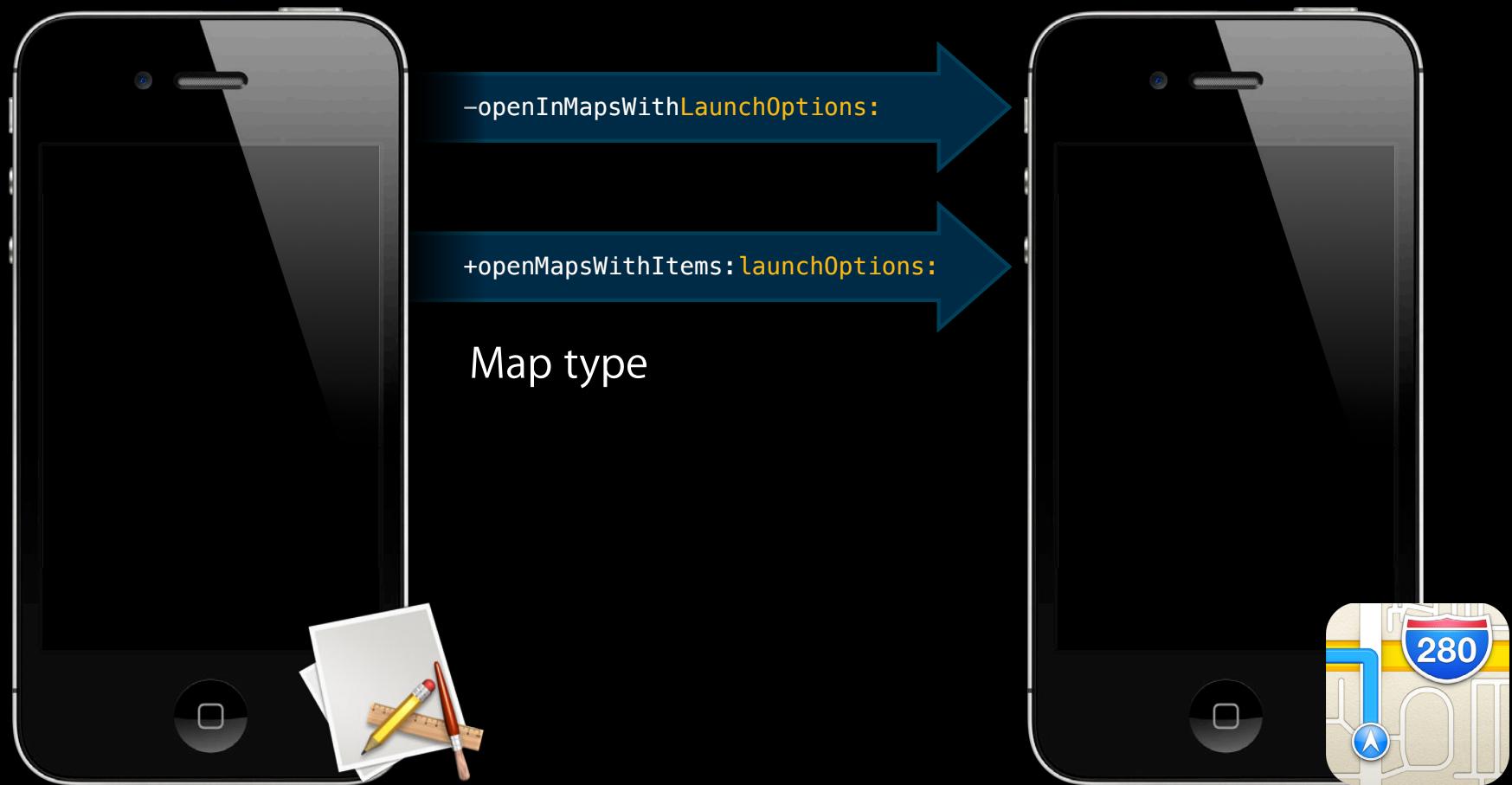
Launch Options



Launch Options



Launch Options



Launch Options



`-openInMapsWithLaunchOptions:`

`+openMapsWithItems:launchOptions:`

Map type

Coordinate region



Launch Options



`-openInMapsWithLaunchOptions:`

`+openMapsWithItems:launchOptions:`

Map type

Coordinate region

Traffic



Launch Options

Map type

```
NSDictionary *options = @{

};

};
```



Launch Options

Map type

```
NSDictionary *options = @{  
    MKLaunchOptionsMapTypeKey :  
        [NSNumber  
            numberWithInt:MKMapTypeSatellite],  
};
```



Launch Options

Coordinate region

```
NSDictionary *options = @{  
    MKLaunchOptionsMapTypeKey :  
        [NSNumber  
            numberWithInt:MKMapTypeSatellite],  
};
```



Launch Options

Coordinate region

```
NSDictionary *options = @{  
    MKLaunchOptionsMapTypeKey :  
        [NSNumber  
            numberWithInt:MKMapTypeSatellite],  
};
```



Launch Options

Coordinate region

```
NSDictionary *options = @{  
    MKLaunchOptionsMapTypeKey :  
        [NSNumber  
            numberWithInt:MKMapTypeSatellite],  
  
    MKLaunchOptionsMapCenterKey :  
        [NSValue  
            valueWithMKCoordinate:regionCenter],  
  
    MKLaunchOptionsMapSpanKey :  
        [NSValue  
            valueWithMKCoordinateSpan:regionSpan]  
};
```



Launch Options

Coordinate region

```
NSDictionary *options = @{  
    MKLaunchOptionsMapTypeKey :  
        [NSNumber  
            numberWithInt:MKMapTypeSatellite],  
  
    MKLaunchOptionsMapCenterKey :  
        [NSValue  
            valueWithMKCoordinate:regionCenter],  
  
    MKLaunchOptionsMapSpanKey :  
        [NSValue  
            valueWithMKCoordinateSpan:regionSpan]  
};
```



Launch Options

Traffic

```
NSDictionary *options = @{

};

};
```



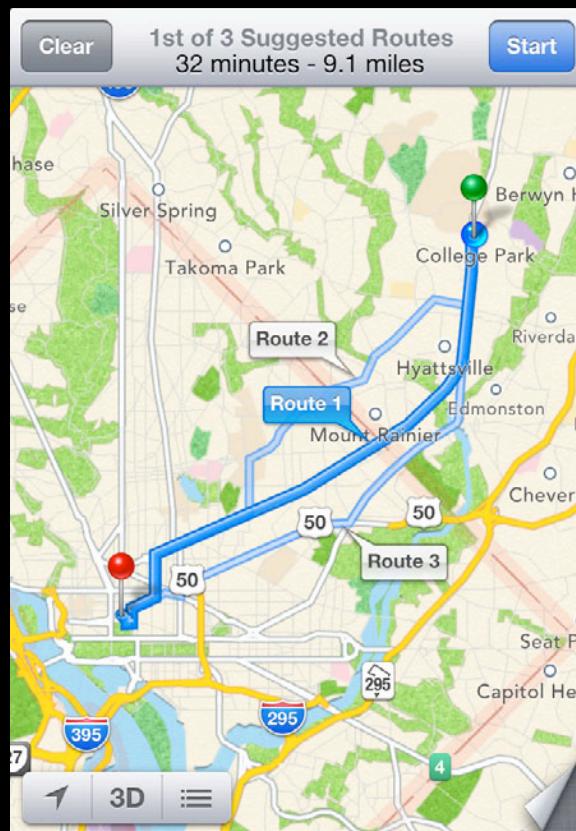
Launch Options

Traffic

```
NSDictionary *options = @{  
  
    MKLaunchOptionsShowsTrafficKey :  
        @YES,  
  
};
```



Getting Directions



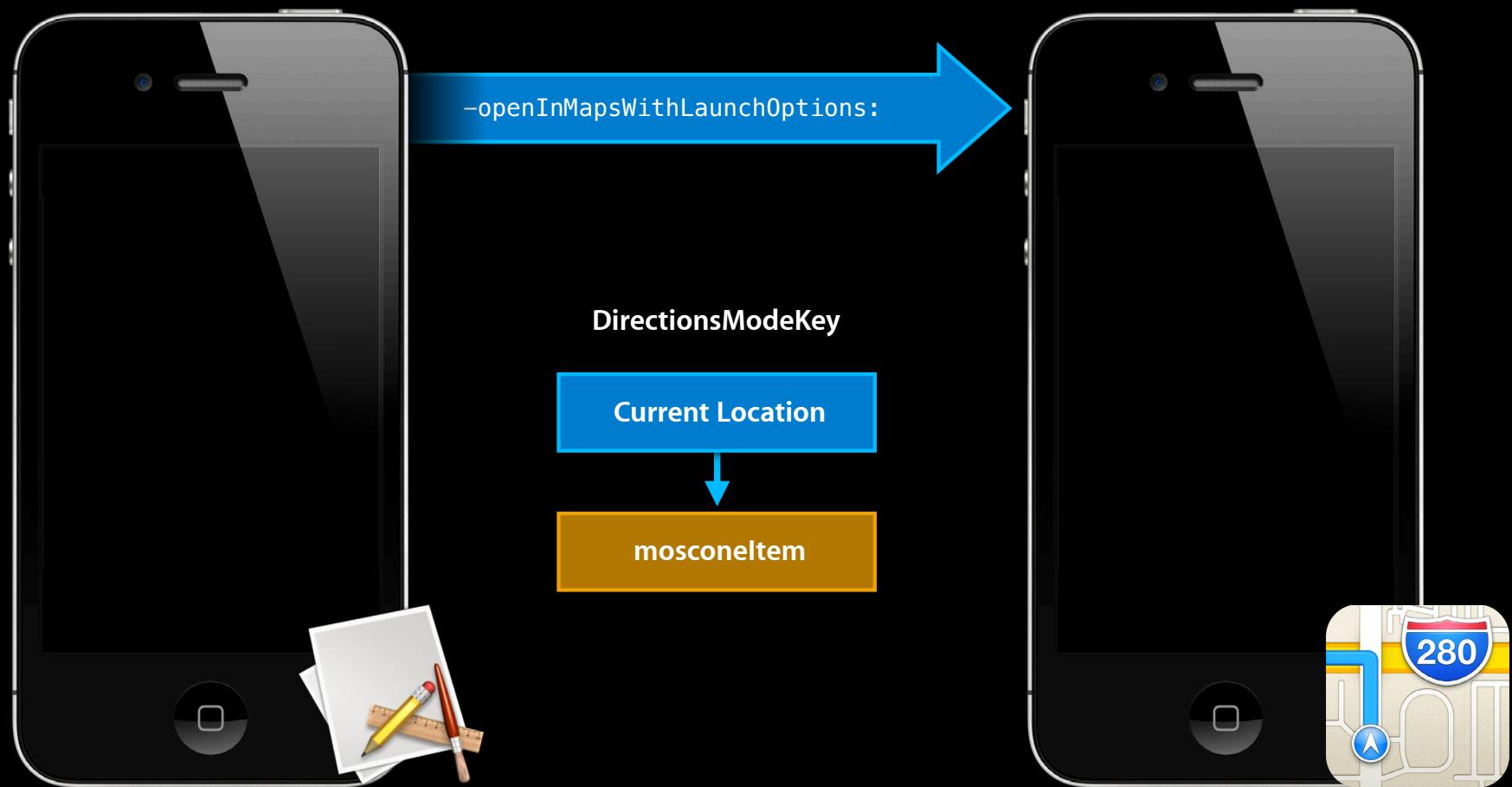
Getting Directions



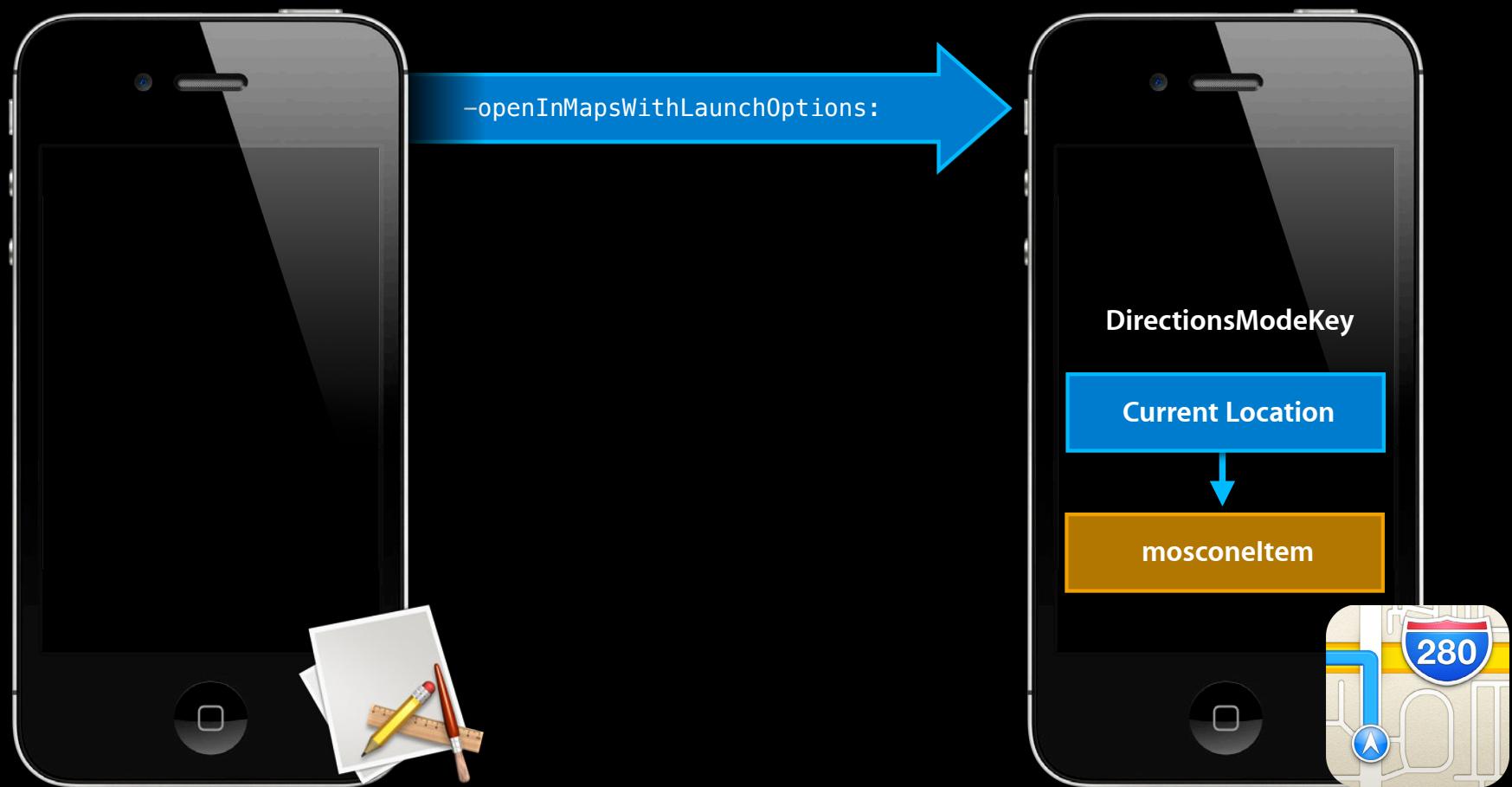
Getting Directions



Getting Directions



Getting Directions



Getting Directions

From your current location

```
MKMapItem *mosconeItem = ...  
  
NSDictionary *options = @{  
    MKLaunchOptionsDirectionsModeKey : MKLaunchOptionsDirectionsModeDriving,  
};  
  
[mosconeItem openInMapsWithLaunchOptions:options];
```

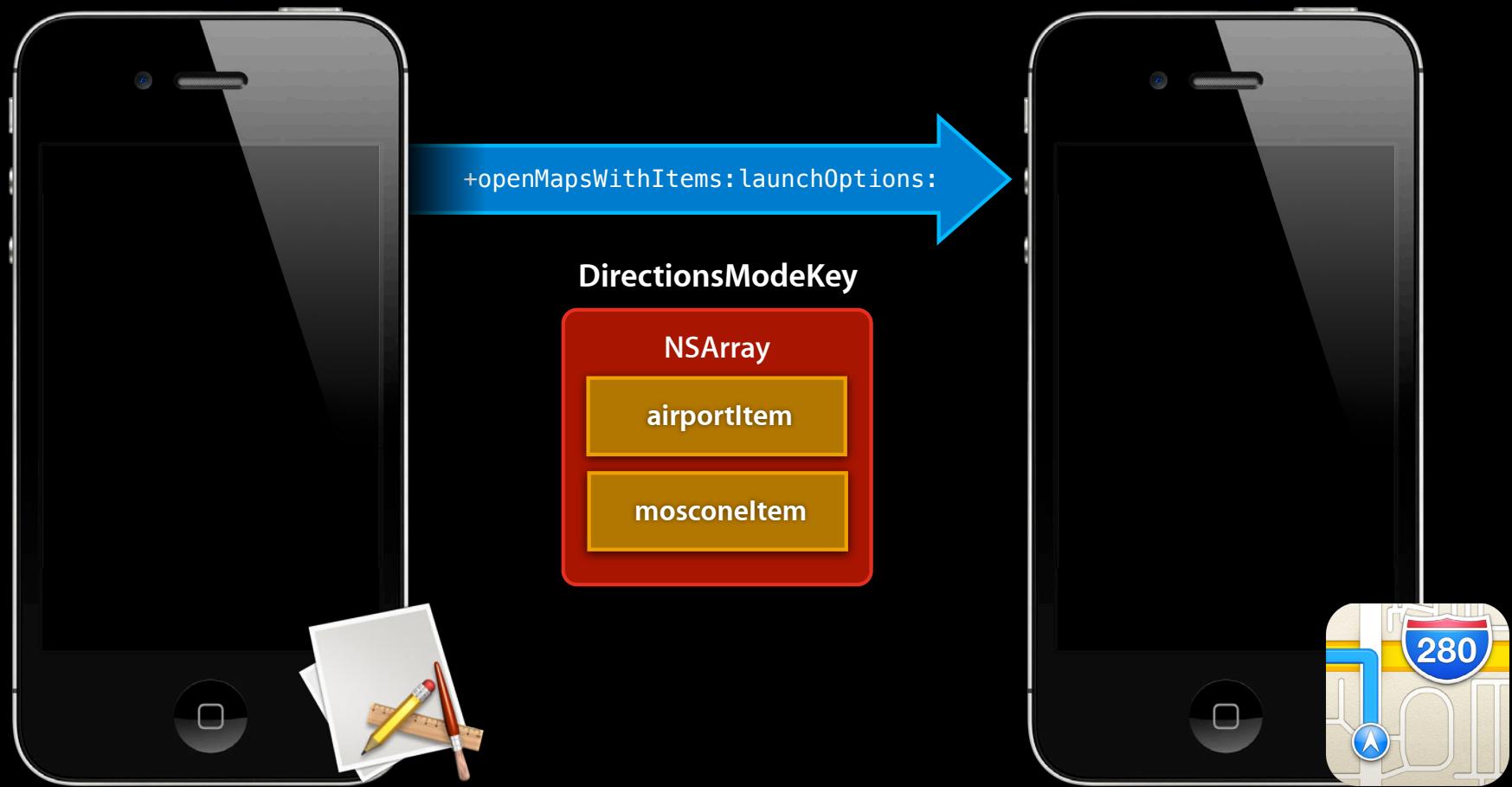
Directions from a Custom Origin



Directions from a Custom Origin



Directions from a Custom Origin



Directions from a Custom Origin



Directions from a Custom Origin



Directions from a Custom Origin

```
NSArray *directionsItems = @[  
    airportItem,  
    mosconeWestItem  
];
```

Directions from a Custom Origin

```
NSArray *directionsItems = @[  
    airportItem,  
    mosconeWestItem  
];
```

Directions from a Custom Origin

```
NSArray *directionsItems = @[  
    airportItem,  
    mosconeWestItem  
];  
  
NSDictionary *options = @{  
    MKLaunchOptionsDirectionsModeKey :  
        MKLaunchOptionsDirectionsModeDriving,  
};
```

Directions from a Custom Origin

```
NSArray *directionsItems = @[  
    airportItem,  
    mosconeWestItem  
];  
  
NSDictionary *options = @{  
    MKLaunchOptionsDirectionsModeKey :  
        MKLaunchOptionsDirectionsModeDriving,  
};  
  
[MKMapItem openMapsWithItems:directionsItems launchOptions:options];
```

Representing Current Location

Representing Current Location

```
MKMapItem *item = [MKMapItem mapItemForCurrentLocation];
```

Representing Current Location

```
MKMapItem *item = [MKMapItem mapItemForCurrentLocation];
```

- Current location item will NOT have a placemark

Representing Current Location

```
MKMapItem *item = [MKMapItem mapItemForCurrentLocation];
```

- Current location item will NOT have a placemark

```
mapItem.isCurrentLocation == YES
```

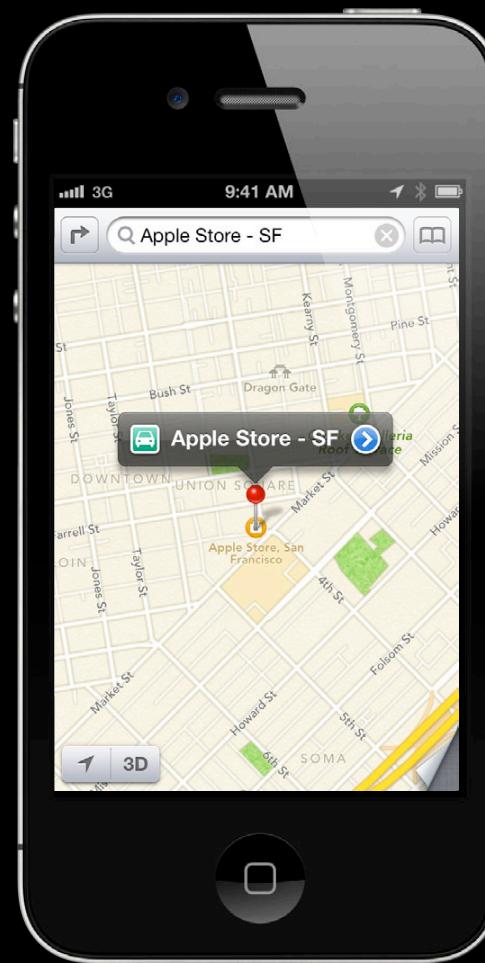
Directions from a Custom Origin

```
NSArray *directionsItems = @[  
    mosconeWestItem,  
    [MKMapItem mapItemForCurrentLocation]  
];  
  
NSDictionary *options = @{  
    MKLaunchOptionsDirectionsModeKey :  
        MKLaunchOptionsDirectionsModeDriving,  
};  
  
[MKMapItem openMapsWithItems:directionsItems launchOptions:options];
```

Launching Maps with MKMapItem

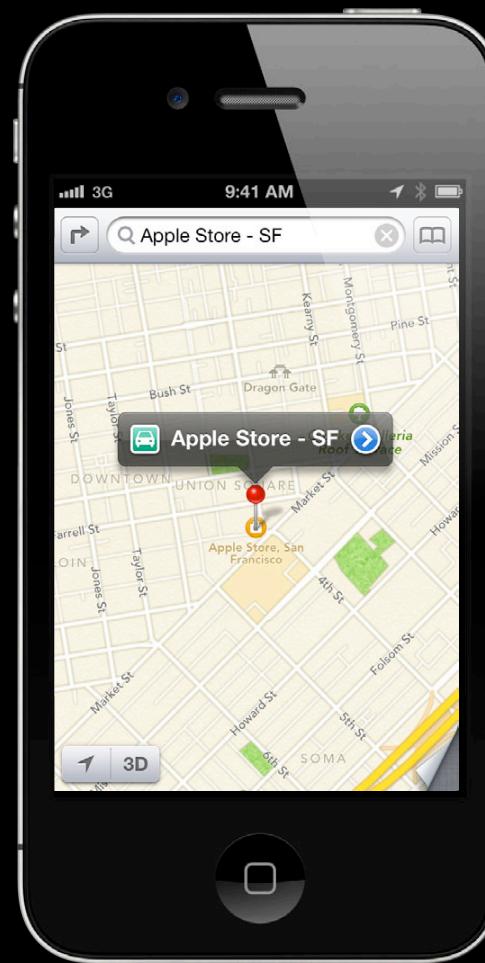
Launching Maps with MKMapItem

- Open Maps with minimal code



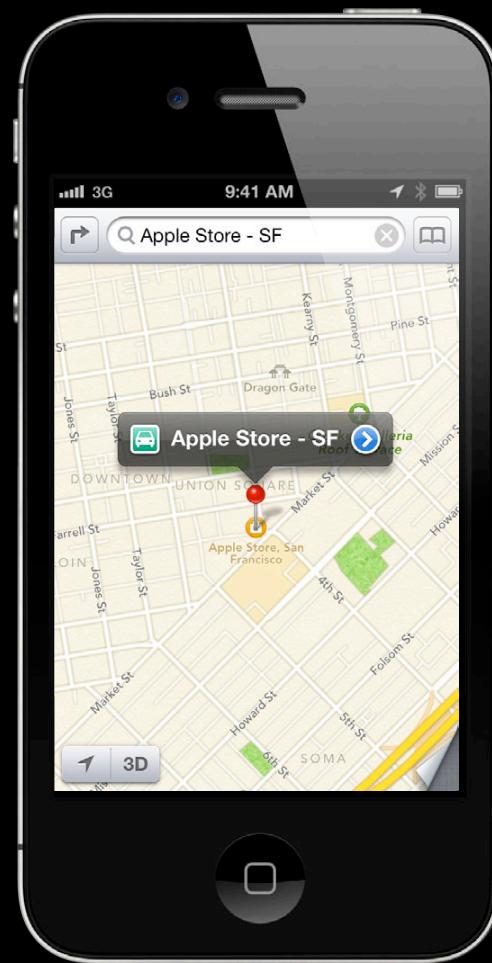
Launching Maps with MKMapItem

- Open Maps with minimal code
- Offer quick access to navigation, bookmarks, and more



Launching Maps with MKMapItem

- Open Maps with minimal code
- Offer quick access to navigation, bookmarks, and more
- Keep your app focused



Developing Routing Apps

Integrated routing powered by the app store

The Many Faces of Routing

Transportation modes

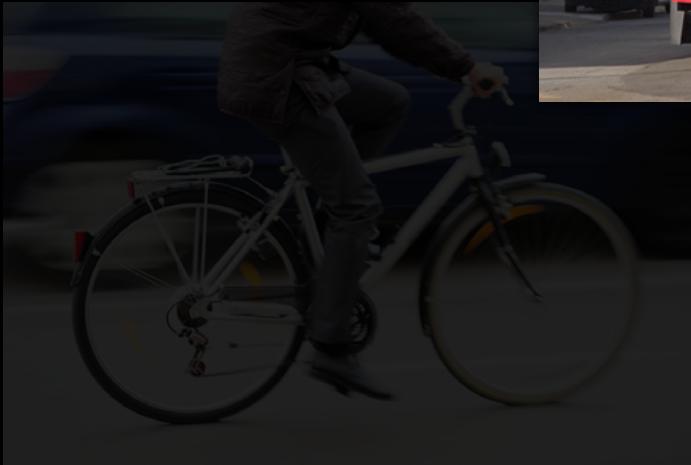
The Many Faces of Routing

Transportation modes



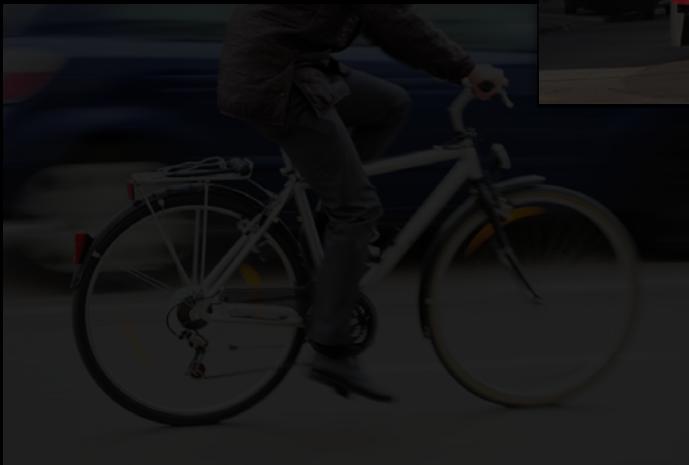
The Many Faces of Routing

Transportation modes



The Many Faces of Routing

Transportation modes







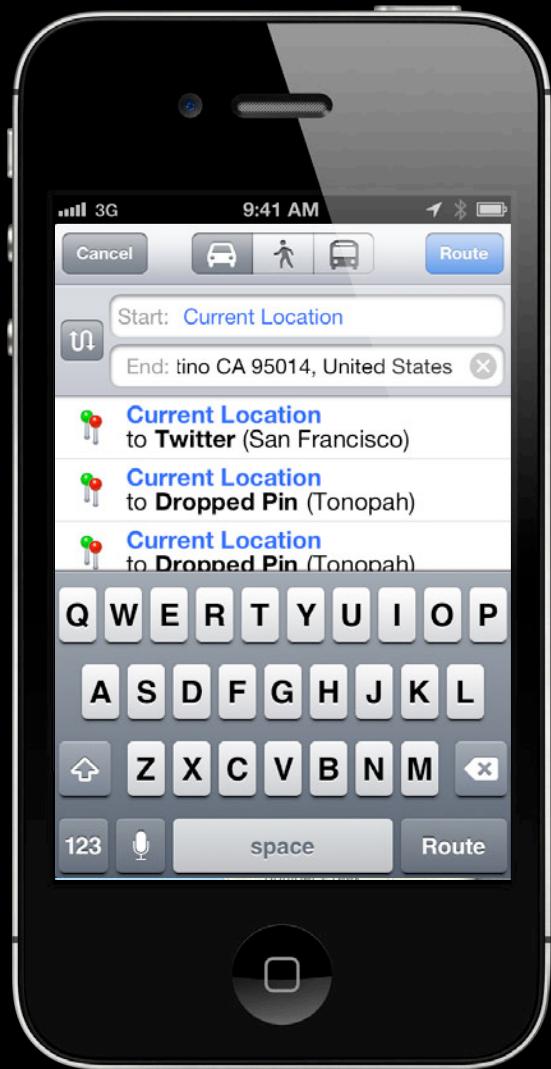
+

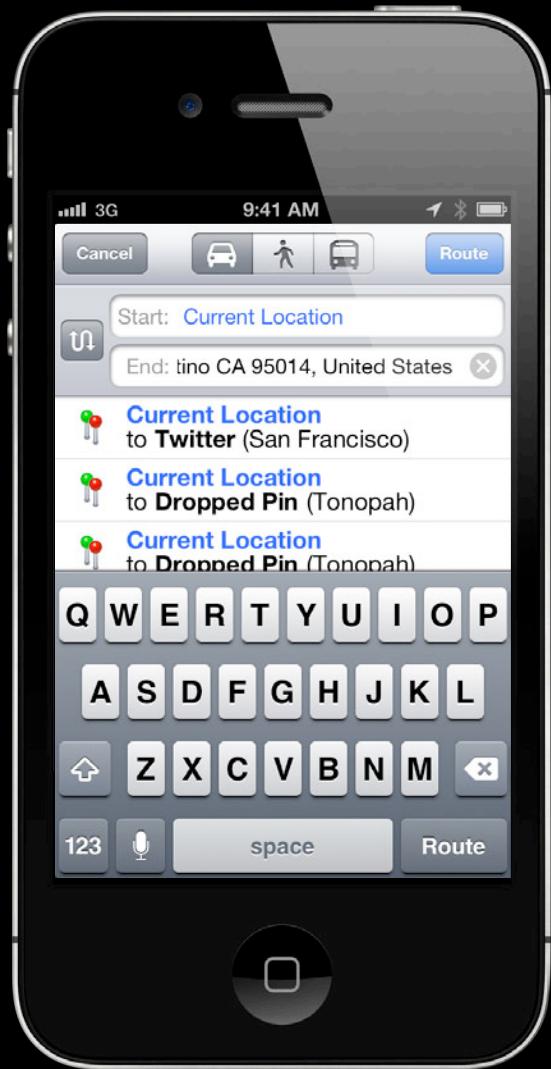


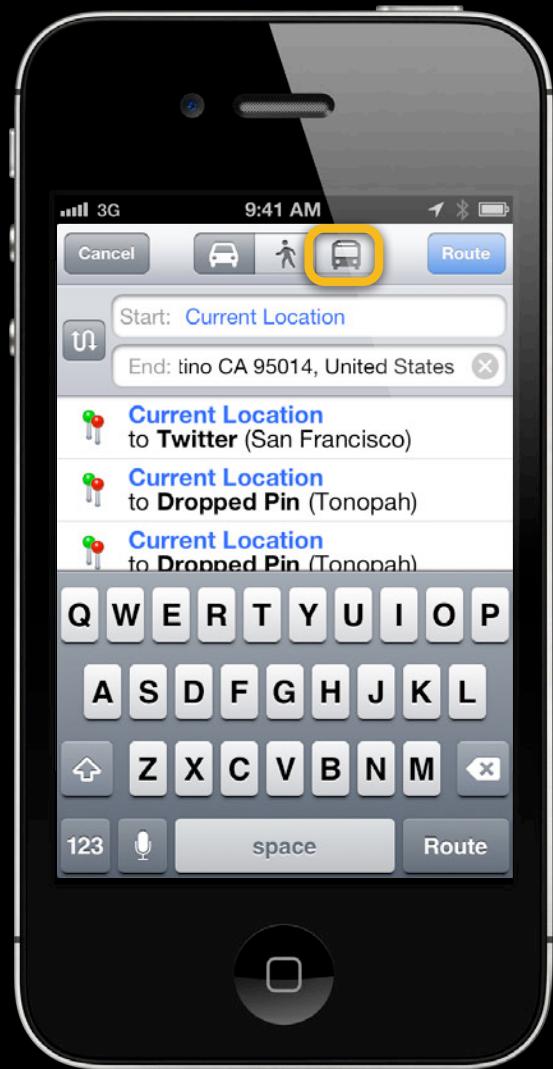


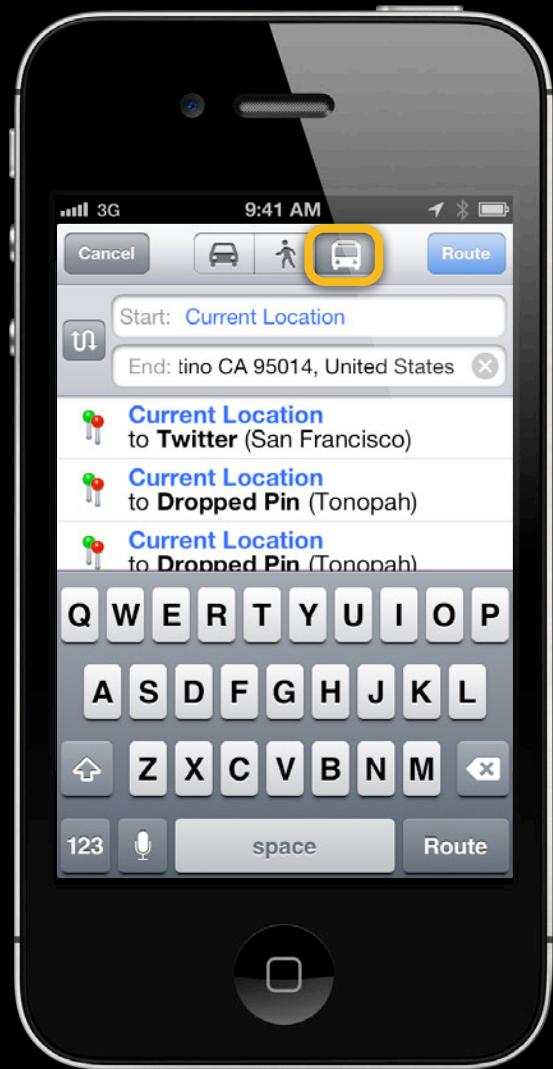


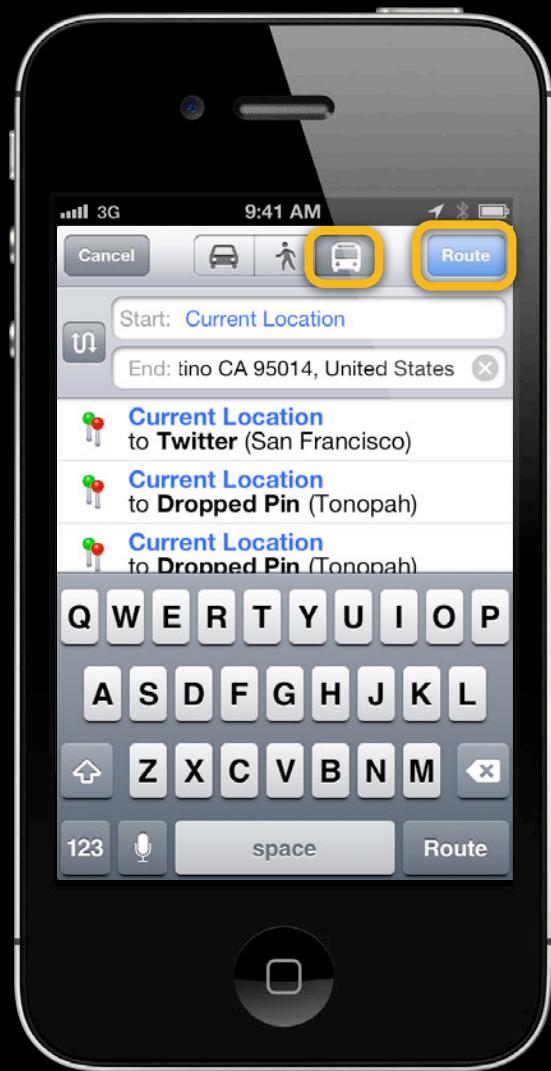


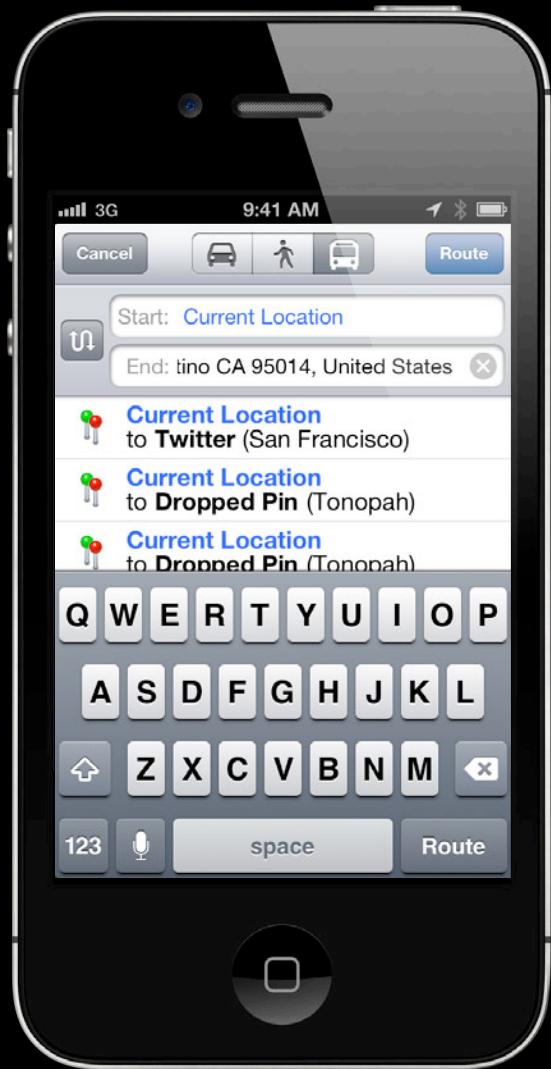


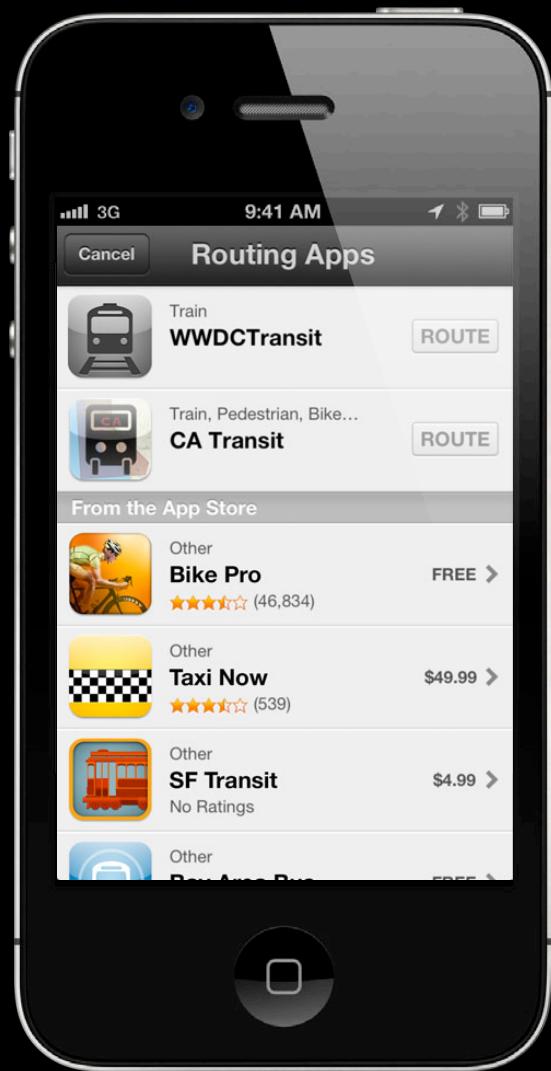


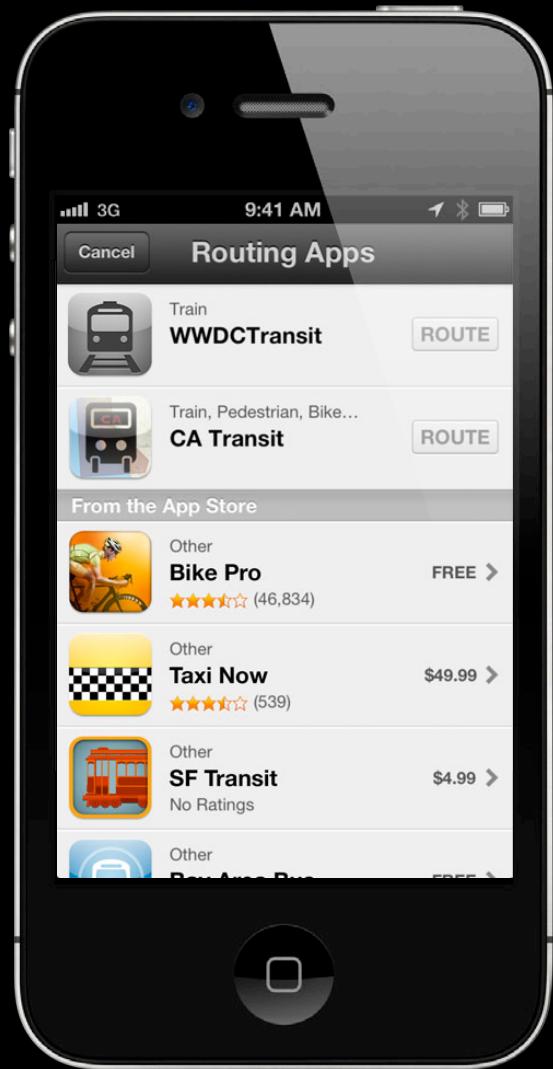


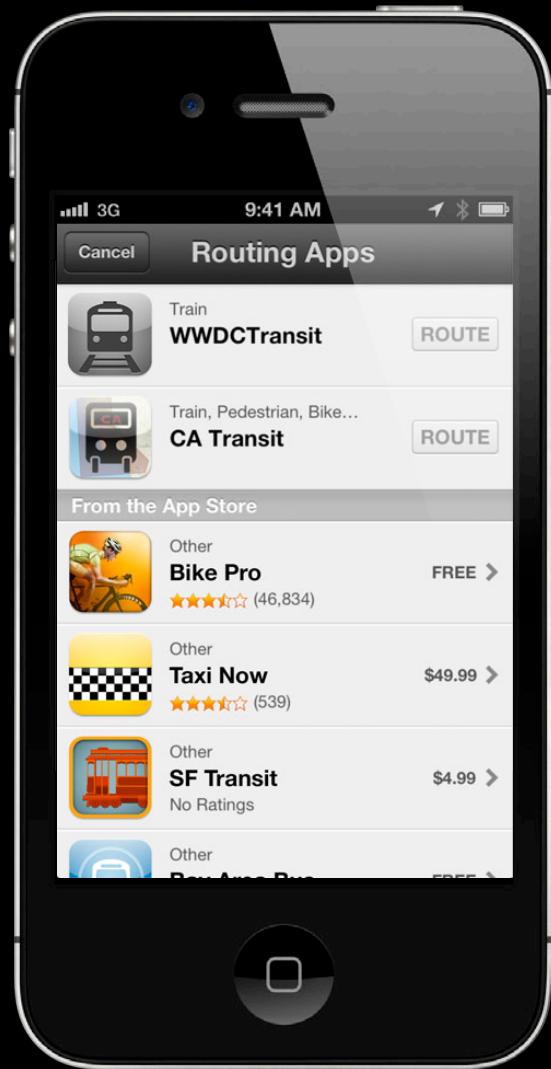




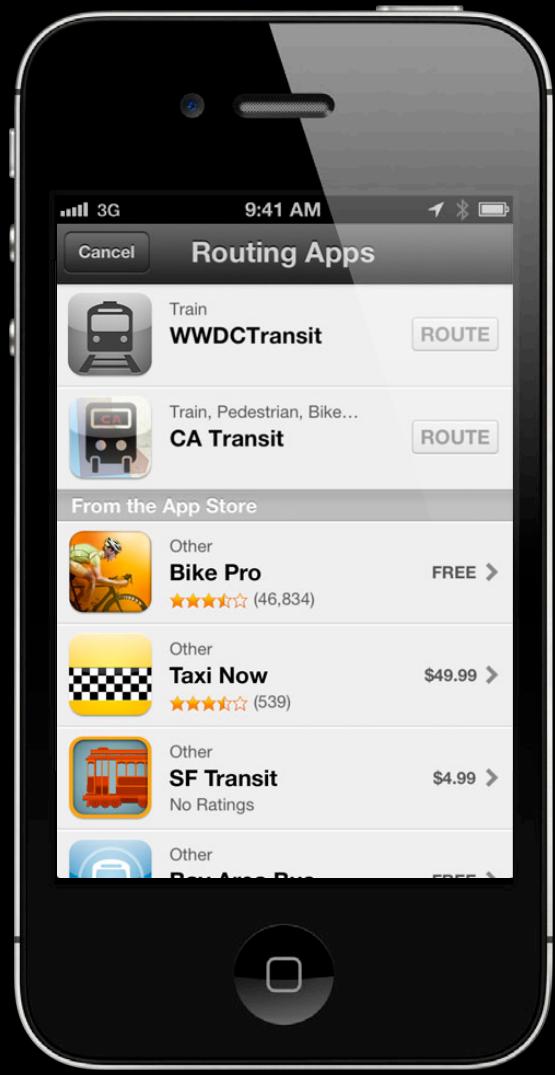




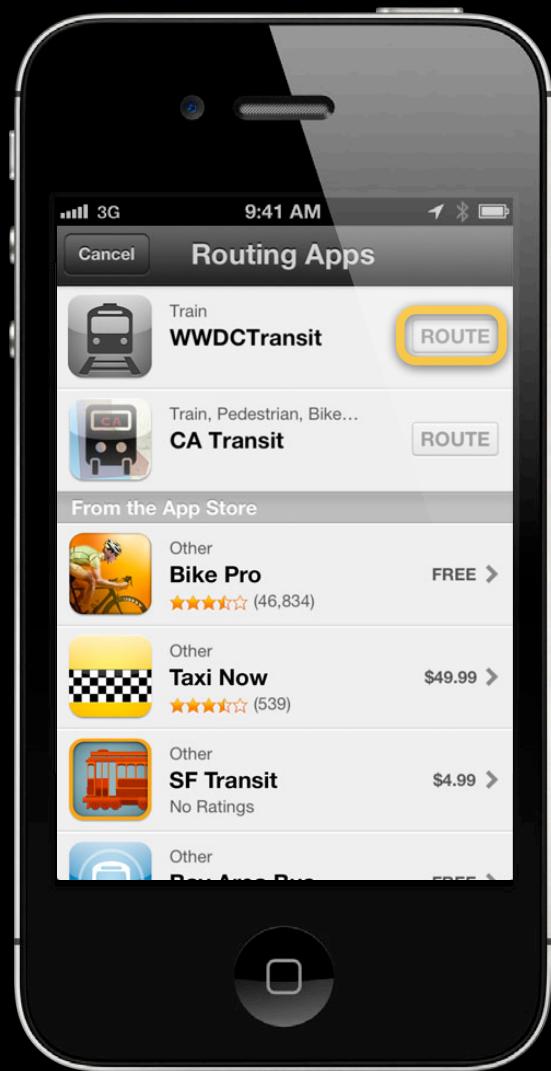


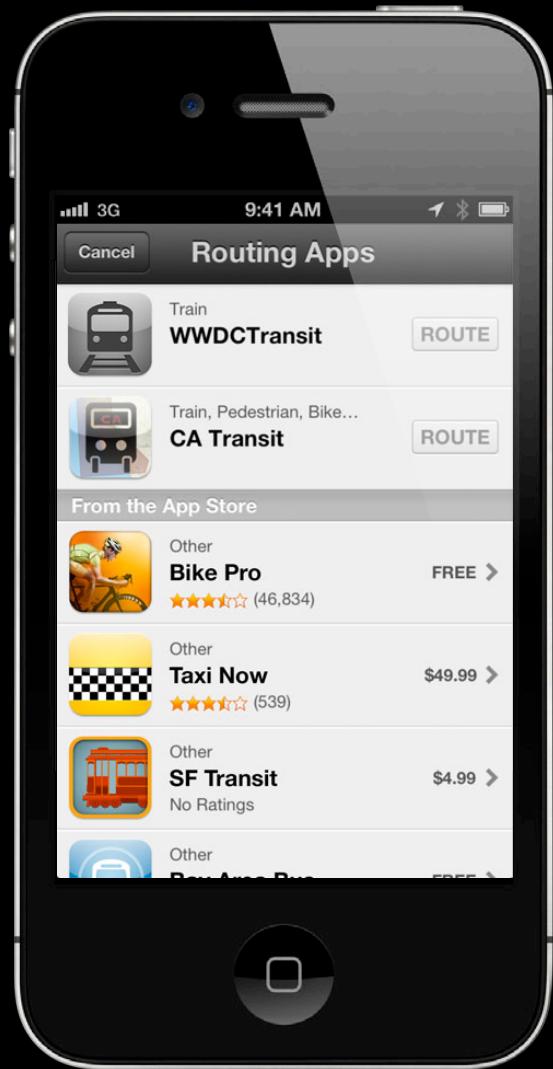


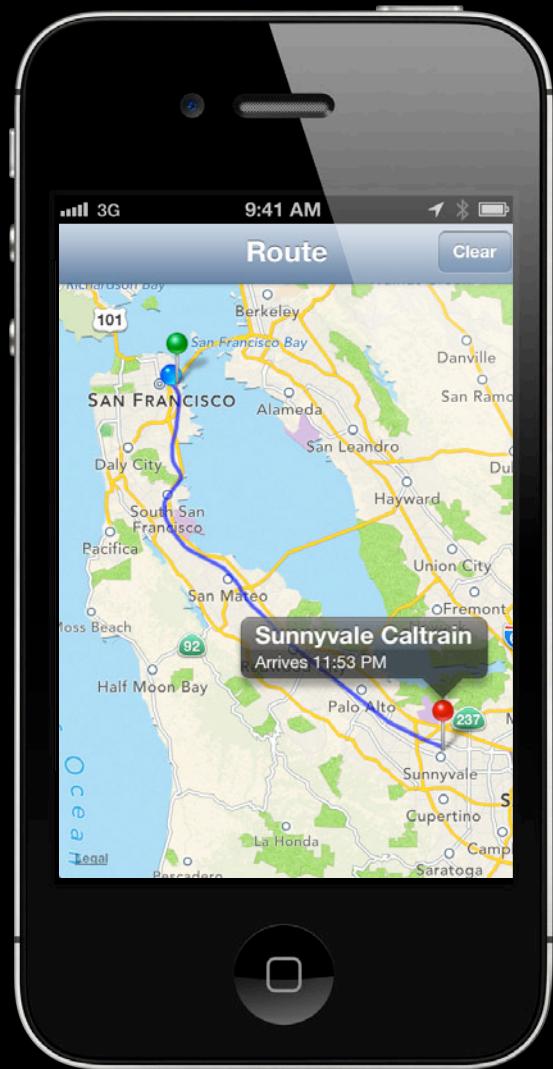
← **Installed Routing Apps**
Recommended apps for your start and endpoint.



← **App Store Suggestions**
Purchase from inside of
Maps.







Routing App Requirements

Routing App Requirements

1. Declare your app as a routing app

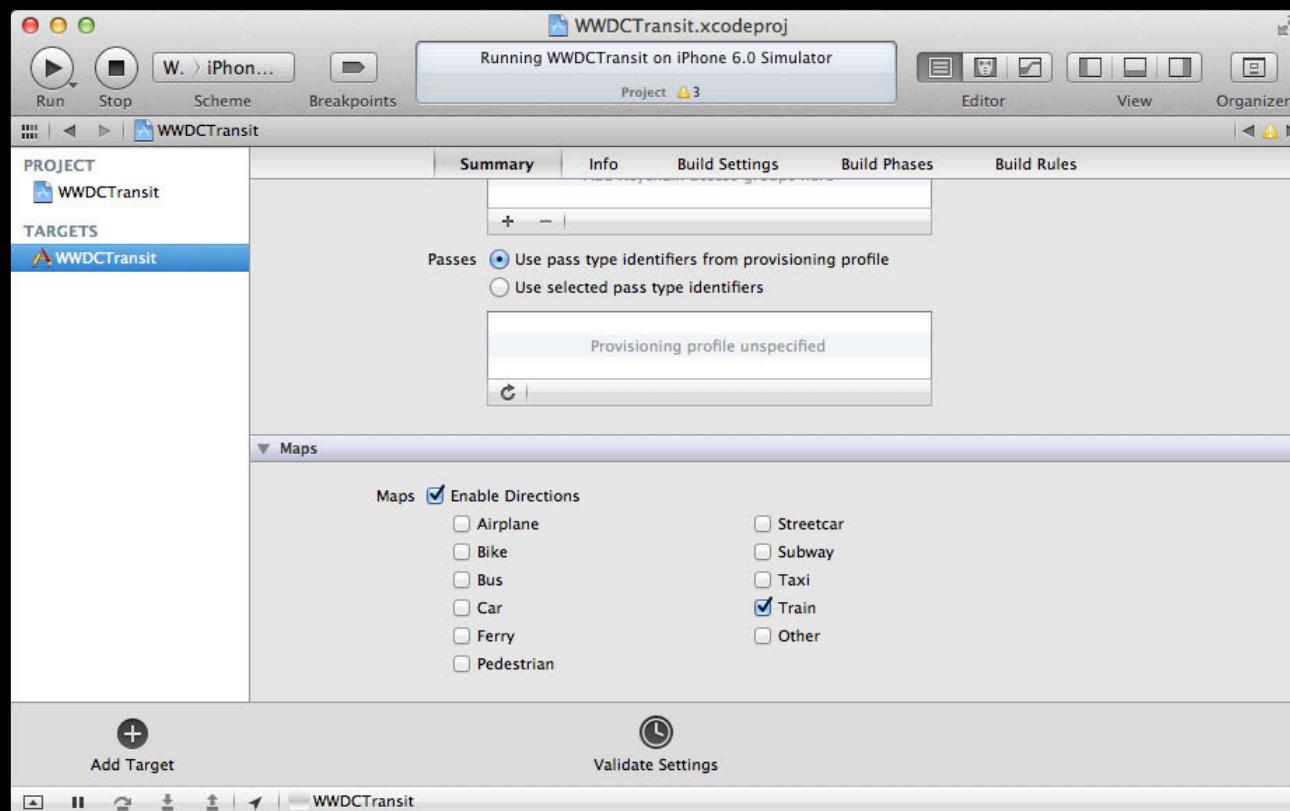
Routing App Requirements

1. Declare your app as a routing app
2. Specify where your app is relevant geographically

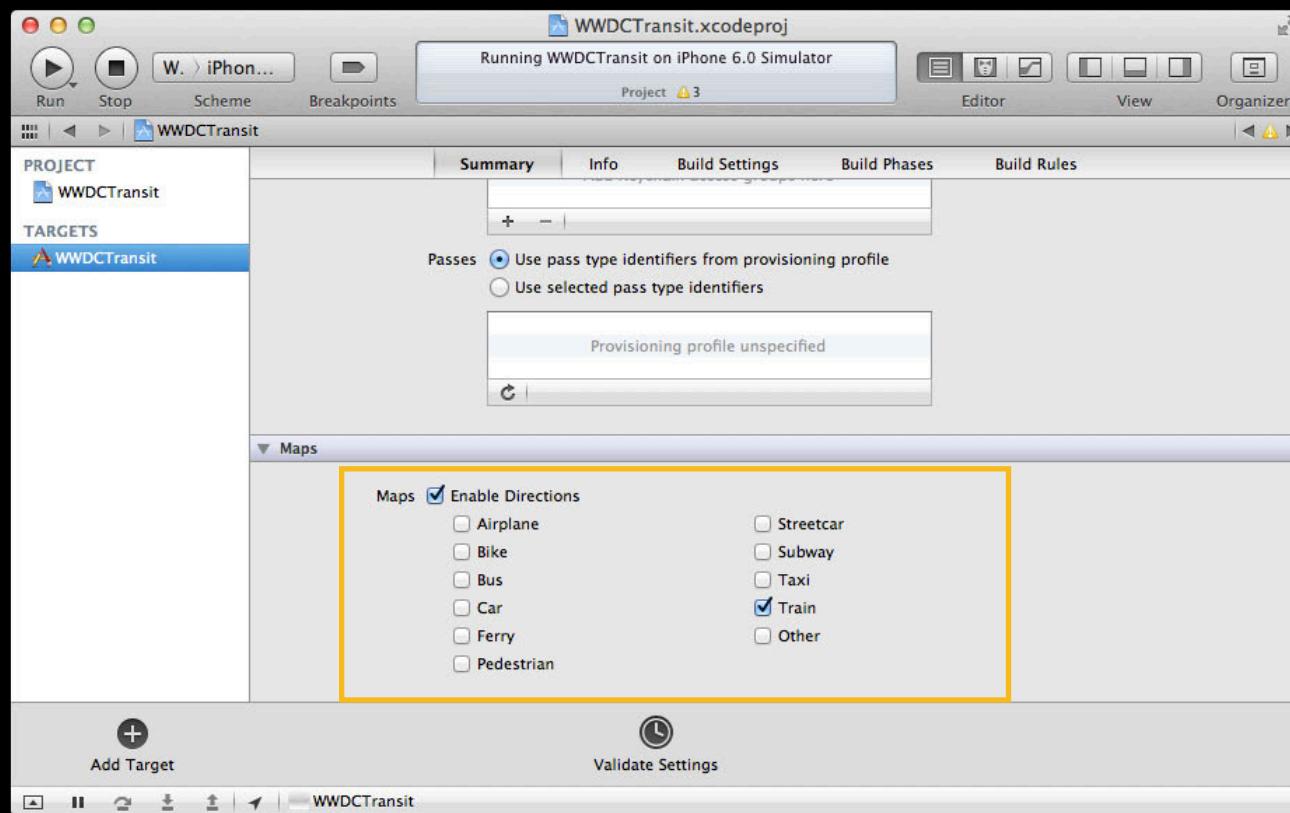
Routing App Requirements

1. Declare your app as a routing app
2. Specify where your app is relevant geographically
3. Handle launches from Maps

Declare Your App as a Routing App



Declare Your App as a Routing App

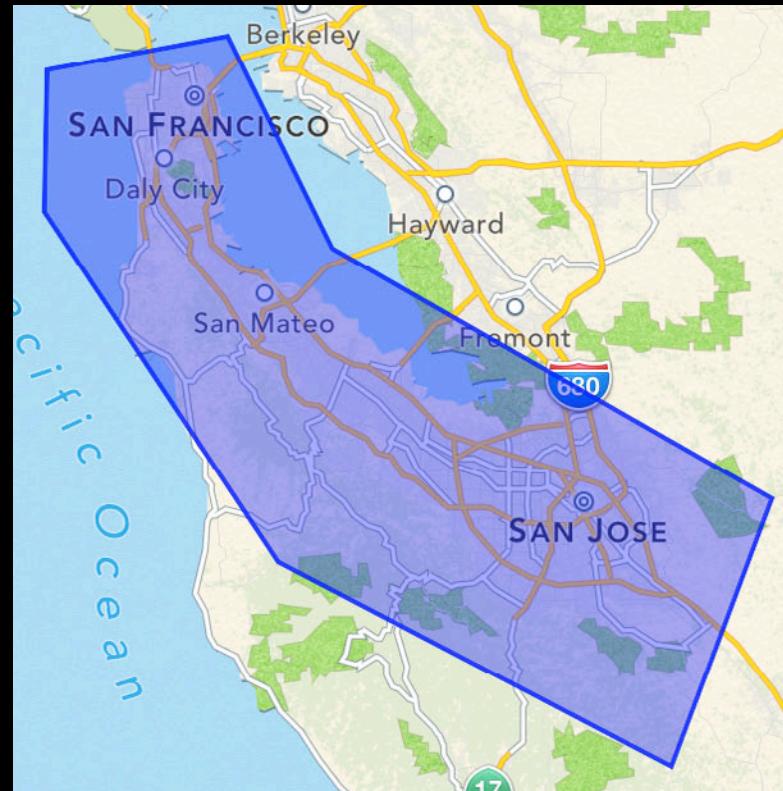


Specifying Geographic Coverage

Where can your app provide routes?

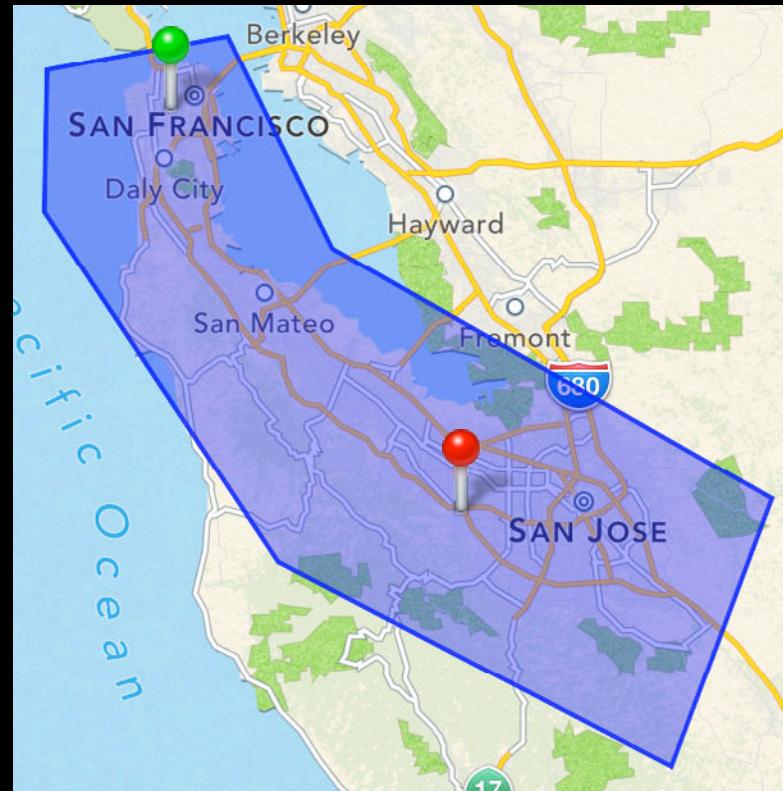
Specifying Geographic Coverage

Where can your app provide routes?



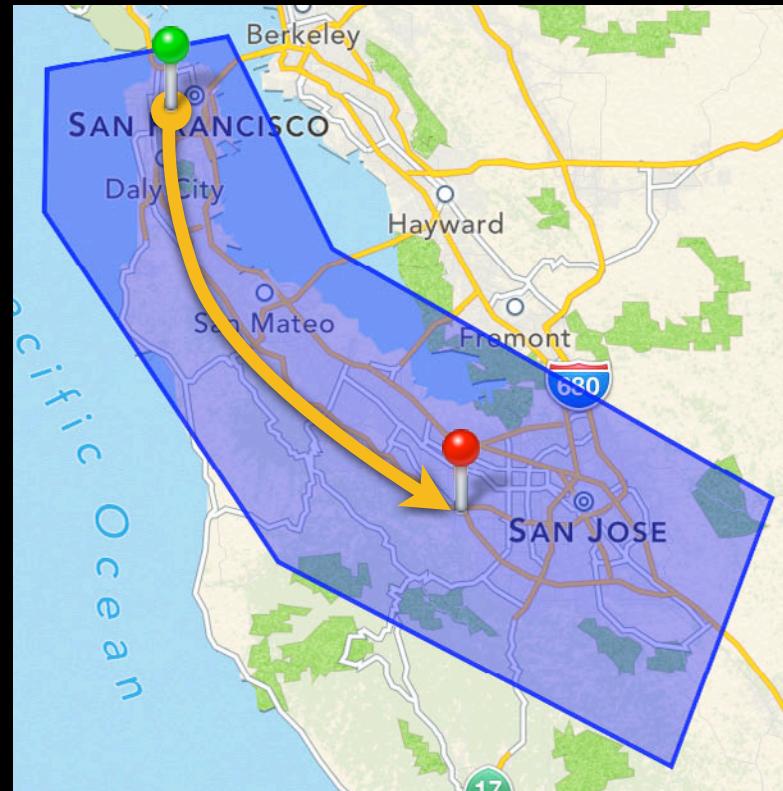
Specifying Geographic Coverage

Where can your app provide routes?



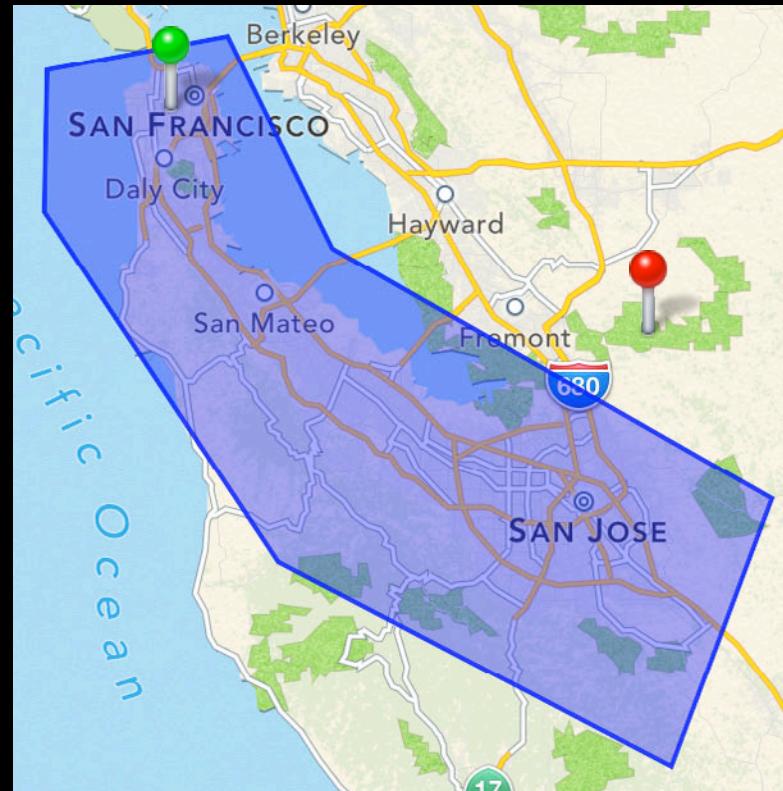
Specifying Geographic Coverage

Where can your app provide routes?



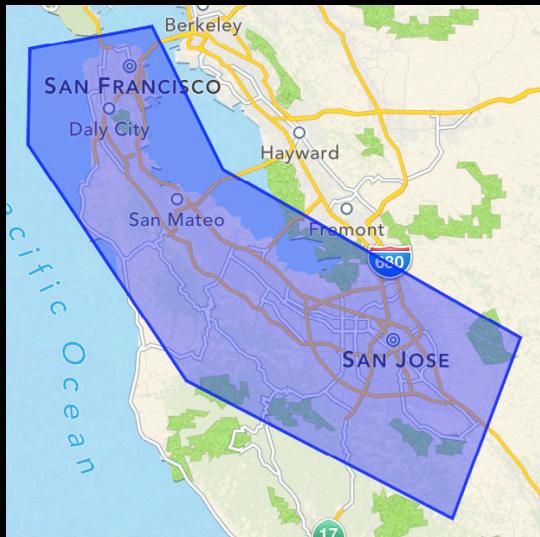
Specifying Geographic Coverage

Where can your app provide routes?



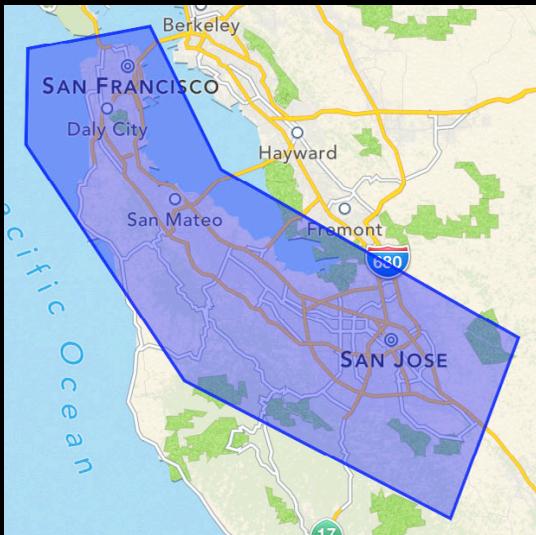
Specifying Geographic Coverage

Where can your app provide routes?



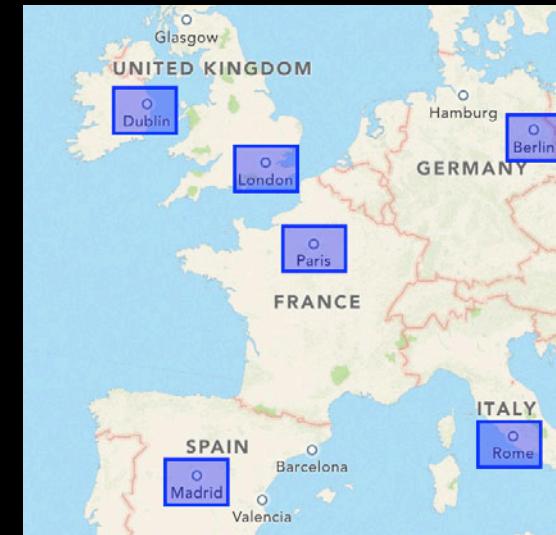
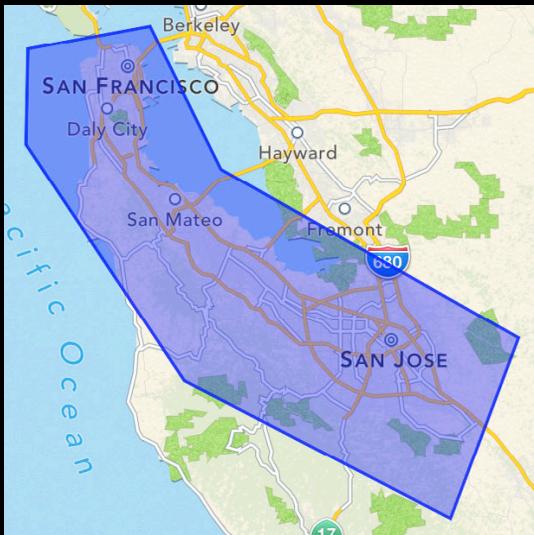
Specifying Geographic Coverage

Where can your app provide routes?



Specifying Geographic Coverage

Where can your app provide routes?



Specifying Geographic Coverage Requirements

Specifying Geographic Coverage Requirements

- GeoJSON file format (.geojson)

Specifying Geographic Coverage Requirements

- GeoJSON file format (.geojson)
- Provide **one** MultiPolygon

Specifying Geographic Coverage Requirements

- GeoJSON file format (.geojson)
- Provide **one** MultiPolygon
- Keep it simple

Specifying Geographic Coverage Requirements

- GeoJSON file format (.geojson)
- Provide **one** MultiPolygon
- Keep it simple
 - Up to **20** polygons, **20** points per polygon

Specifying Geographic Coverage Requirements

- GeoJSON file format (.geojson)
- Provide **one** MultiPolygon
- Keep it simple
 - Up to **20** polygons, **20** points per polygon
- Coverage file is **NOT** part of the app bundle

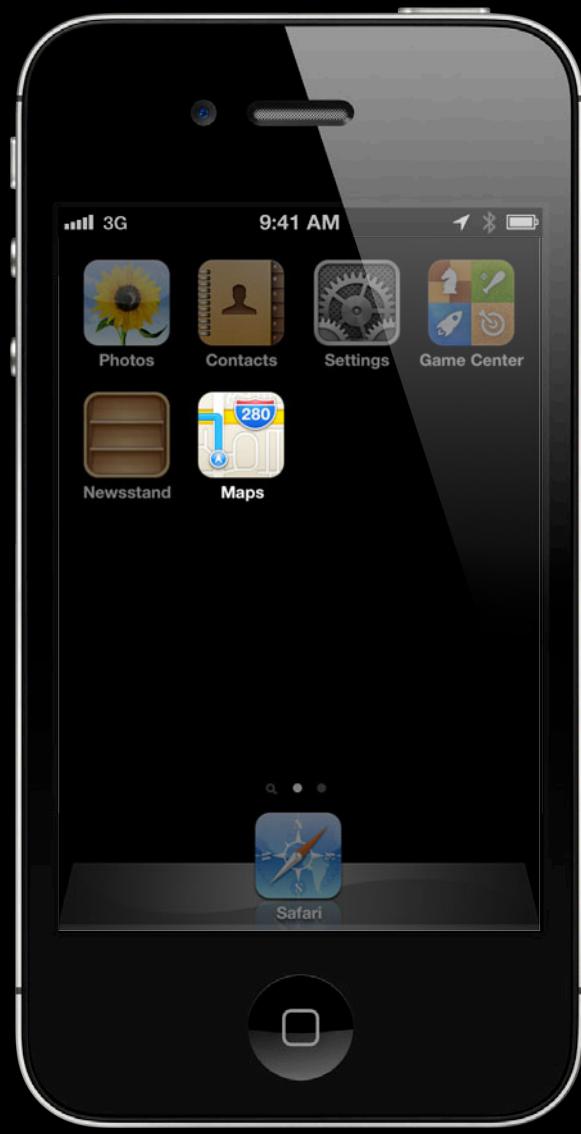
Specifying Geographic Coverage Requirements

- GeoJSON file format (.geojson)
- Provide **one** MultiPolygon
- Keep it simple
 - Up to **20** polygons, **20** points per polygon
- Coverage file is **NOT** part of the app bundle
 - Uploaded using iTunes Connect

Specifying Geographic Coverage Requirements

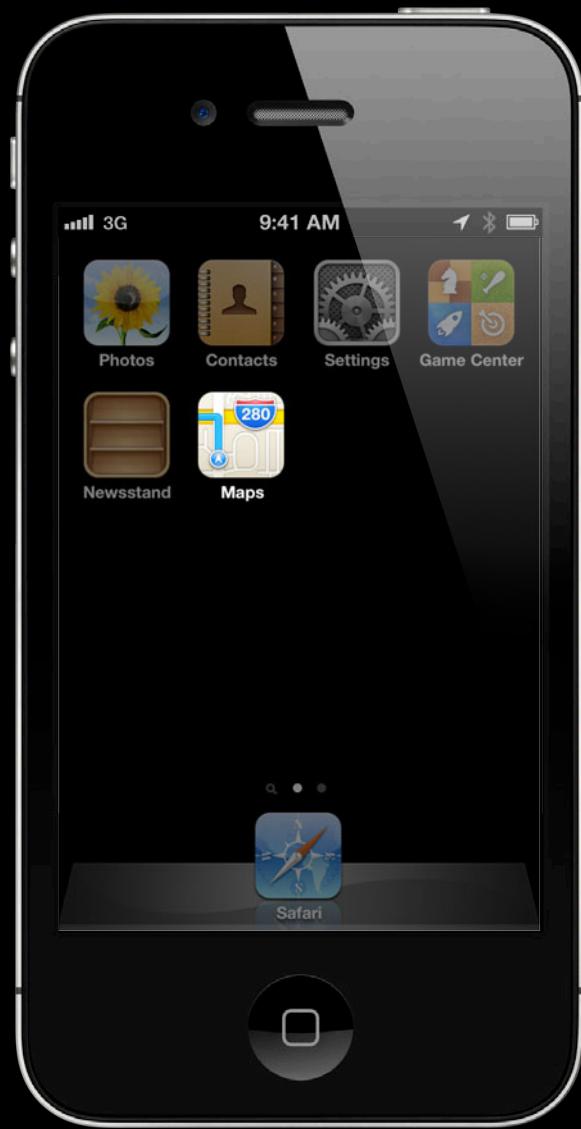
- GeoJSON file format (.geojson)
- Provide **one** MultiPolygon
- Keep it simple
 - Up to **20** polygons, **20** points per polygon
- Coverage file is **NOT** part of the app bundle
 - Uploaded using iTunes Connect
 - Automatically updated

Testing Your Routing App



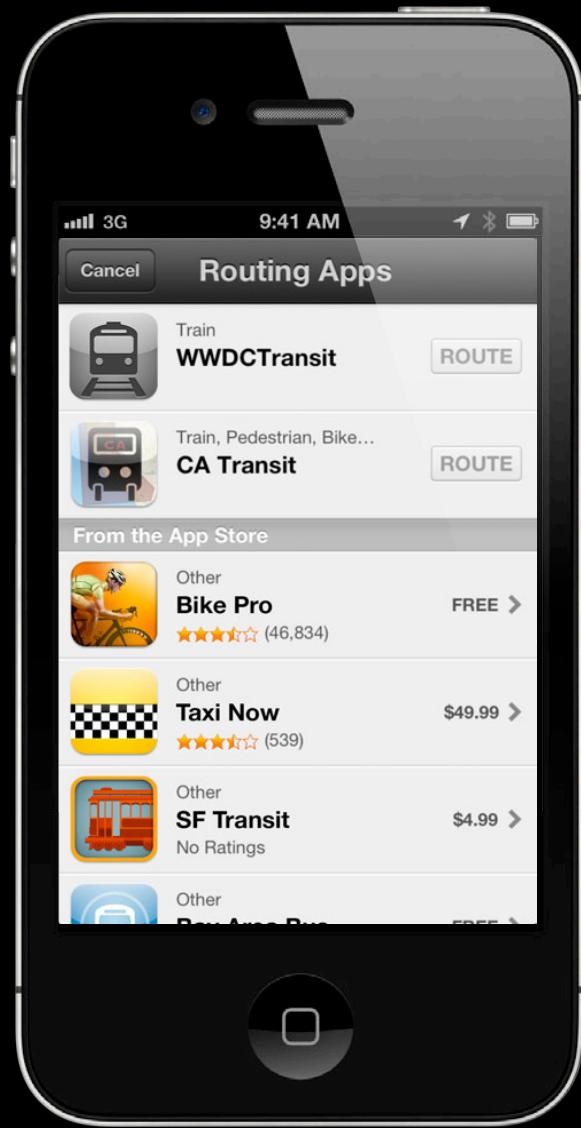
Testing Your Routing App

- Maps is now in the simulator



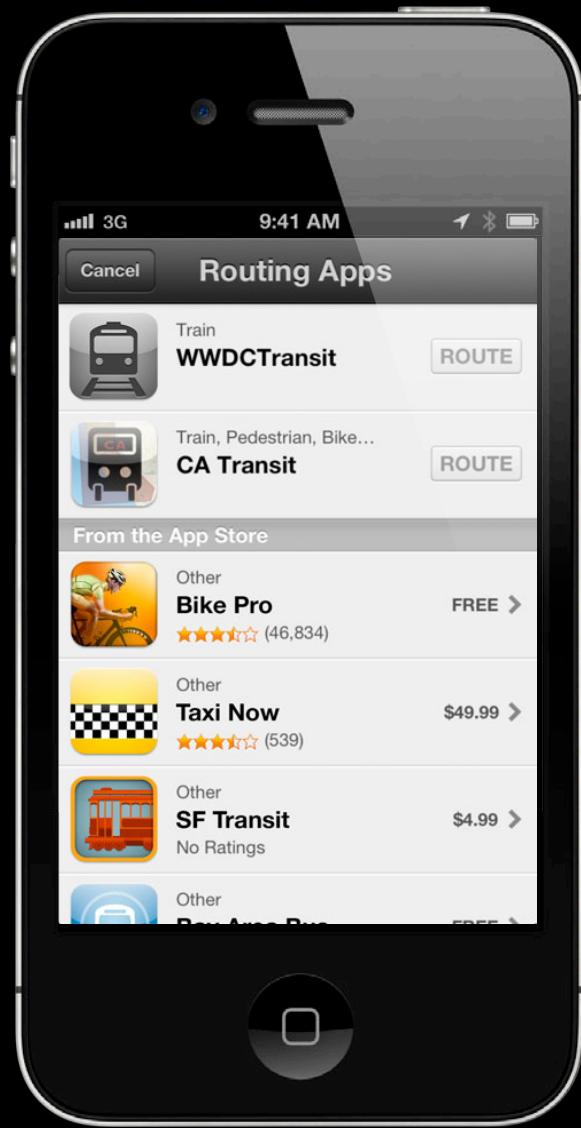
Testing Your Routing App

- Maps is now in the simulator
- Verify your app appears when relevant



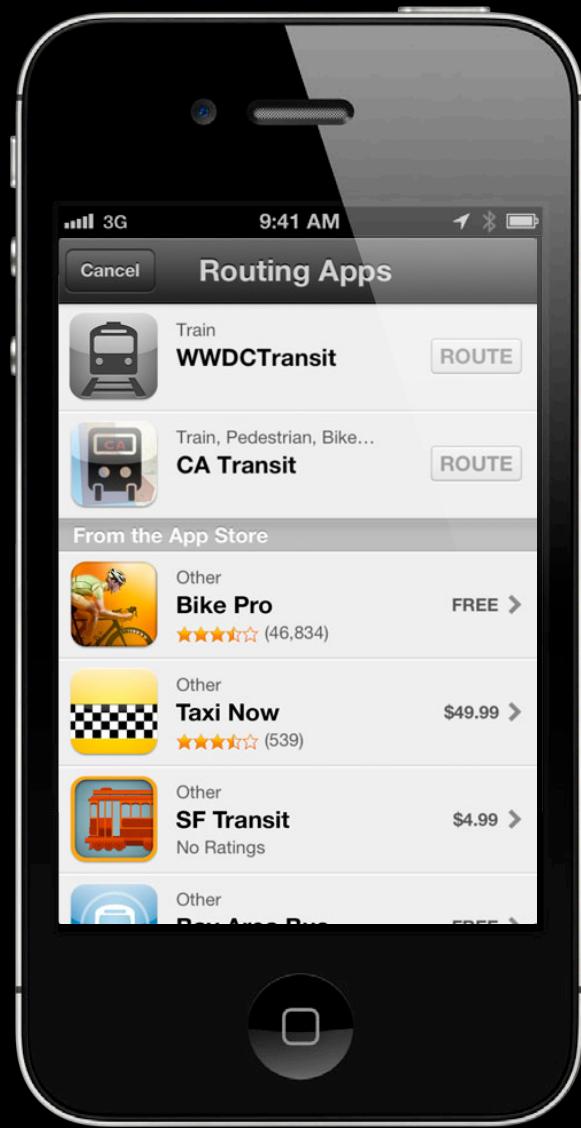
Testing Your Routing App

- Maps is now in the simulator
- Verify your app appears when relevant
- Check the logs



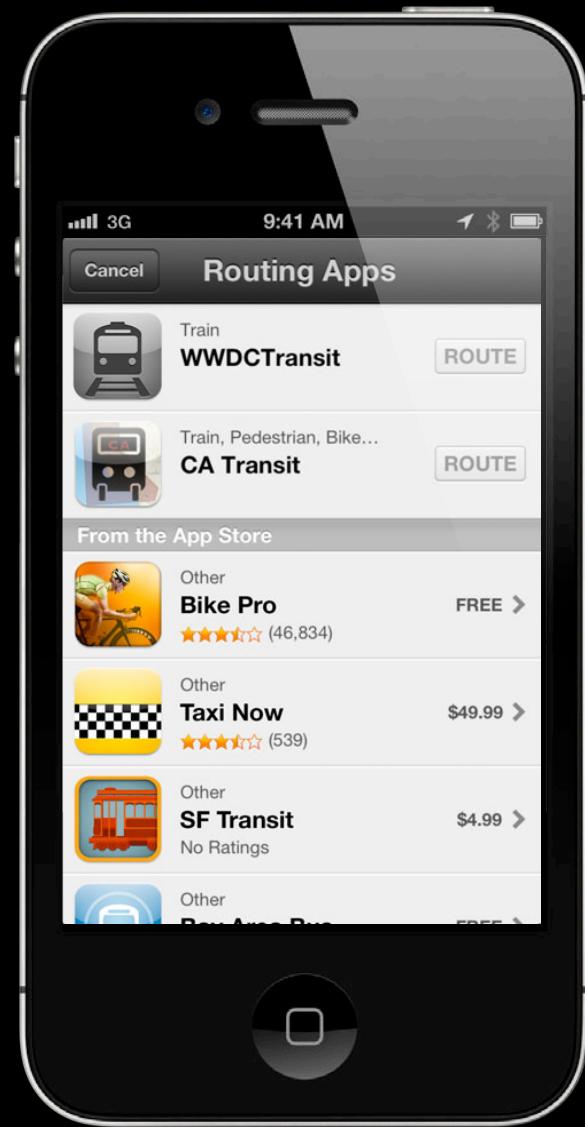
Testing Your Routing App

- Maps is now in the simulator
- Verify your app appears when relevant
- Check the logs
 - Simulator—use Console



Testing Your Routing App

- Maps is now in the simulator
- Verify your app appears when relevant
- Check the logs
 - Simulator—use Console
 - Device—use Xcode Organizer



Demo

Declaring your app as a routing app

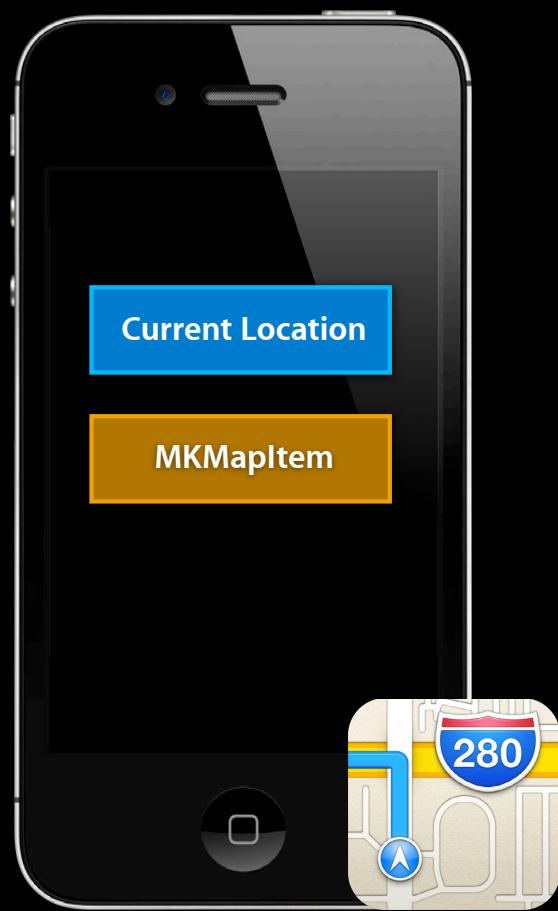
Routing App Requirements

1. Declare your app as a routing app
2. Specify where your app is relevant geographically
3. Handle launches from Maps

Handling Launches from Maps



Handling Launches from Maps



Handling Launches from Maps



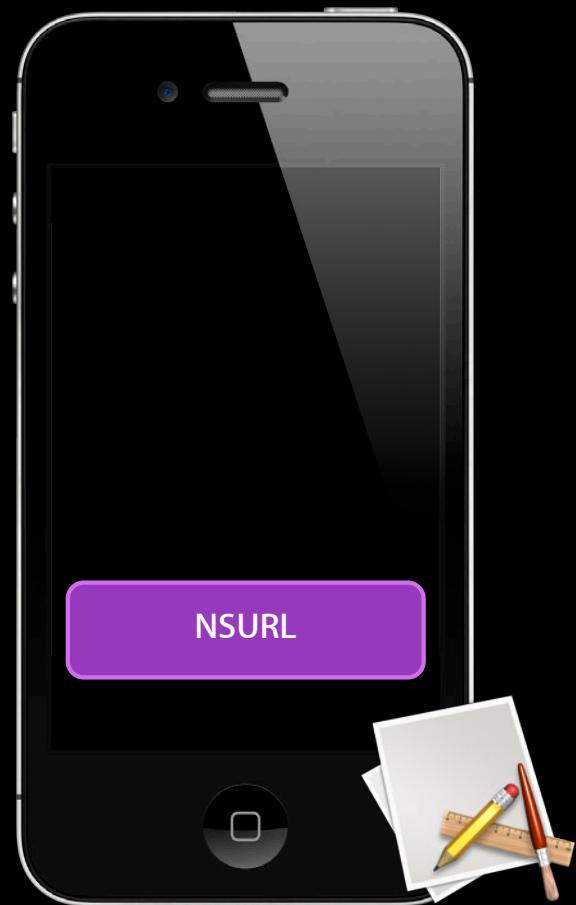
Handling Launches from Maps



Handling Launches from Maps



Handling Launches from Maps



Handling Launches from Maps

- Is it a MKDirectionsRequest URL?



Handling Launches from Maps

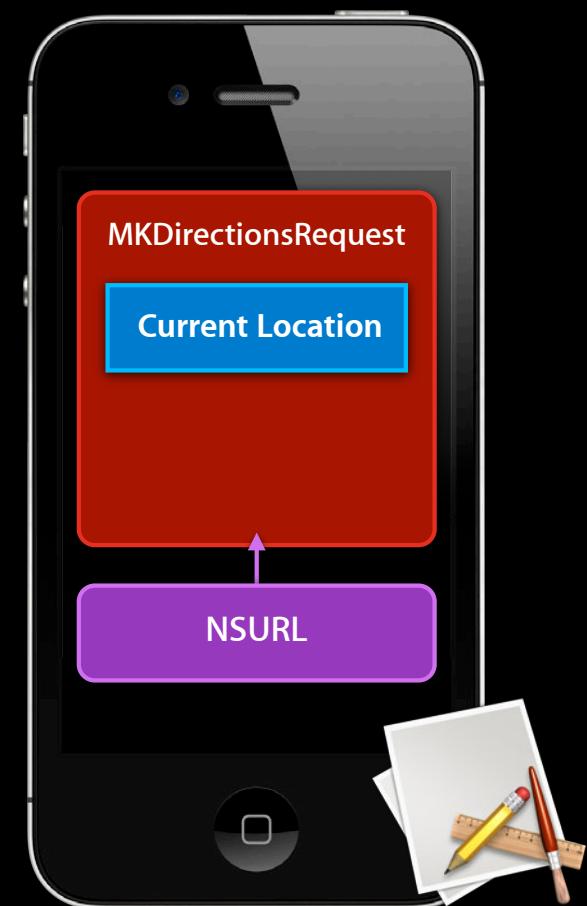
- Is it a MKDirectionsRequest URL?
- If yes, instantiate the request using the URL



Handling Launches from Maps

- Is it a MKDirectionsRequest URL?
- If yes, instantiate the request using the URL

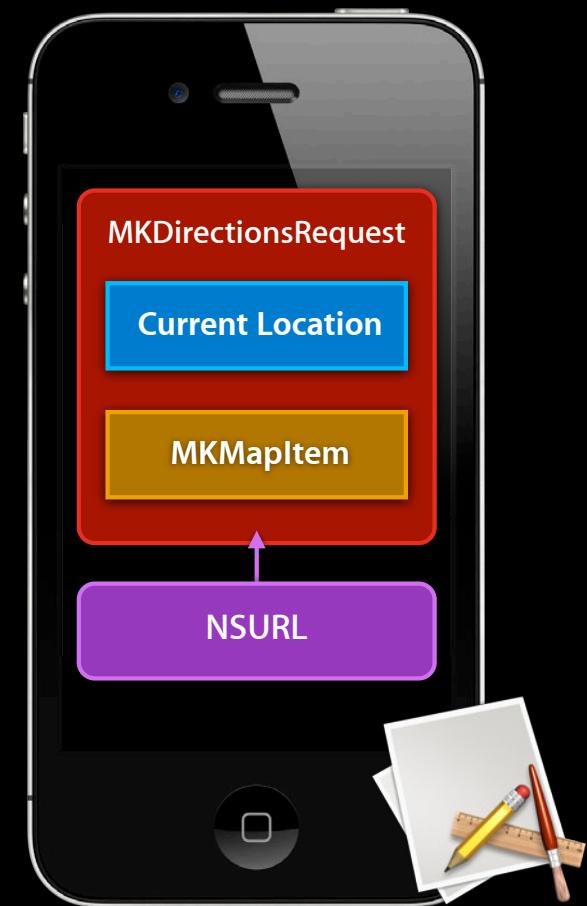
```
@property MKMapItem *source;
```



Handling Launches from Maps

- Is it a MKDirectionsRequest URL?
- If yes, instantiate the request using the URL

```
@property MKMapItem *source;  
@property MKMapItem *destination;
```

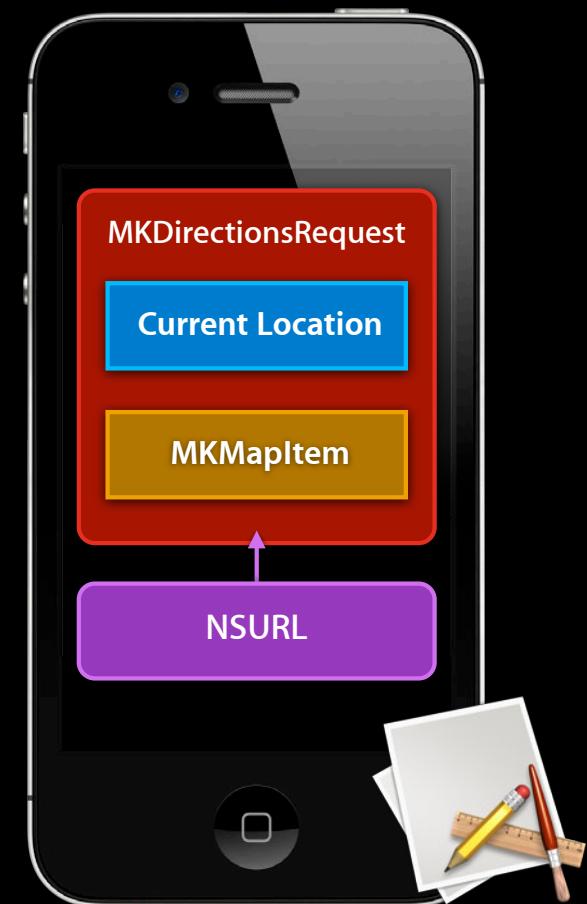


Handling Launches from Maps

- Is it a MKDirectionsRequest URL?
- If yes, instantiate the request using the URL

```
@property MKMapItem *source;  
@property MKMapItem *destination;
```

- Resolve the current location map item

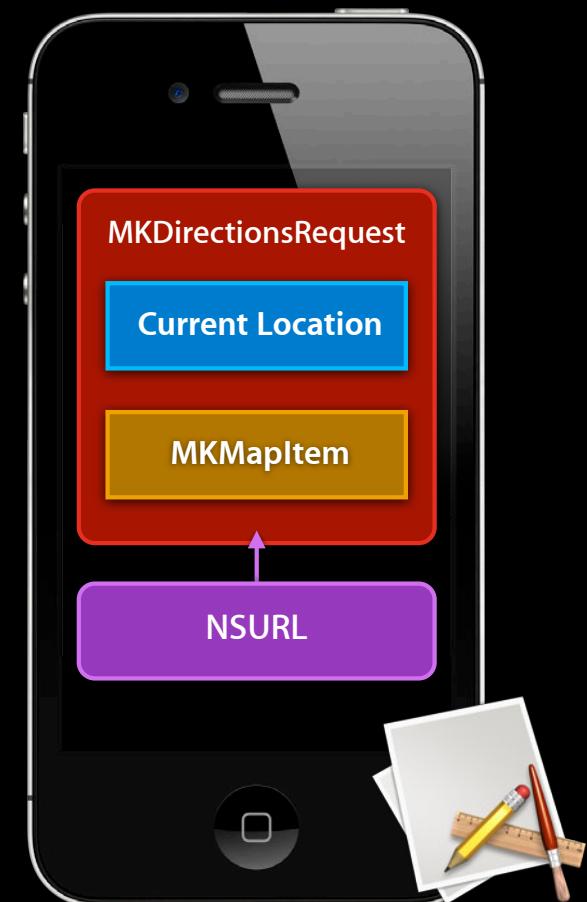


Handling Launches from Maps

- Is it a MKDirectionsRequest URL?
- If yes, instantiate the request using the URL

```
@property MKMapItem *source;  
@property MKMapItem *destination;
```

- Resolve the current location map item
- Provide your route!



Handling Launches from Maps

```
- (BOOL)application:(UIApplication *)application  
    openURL:(NSURL *)url  
sourceApplication:(NSString *)sourceApplication  
annotation:(id)annotation  
{  
  
}
```

Handling Launches from Maps

```
- (BOOL)application:(UIApplication *)application
    openURL:(NSURL *)url
sourceApplication:(NSString *)sourceApplication
annotation:(id)annotation
{
    if ([MKDirectionsRequest isDirectionsRequestURL:url]) {
        }
}
```

Handling Launches from Maps

```
- (BOOL)application:(UIApplication *)application
              openURL:(NSURL *)url
            sourceApplication:(NSString *)sourceApplication
              annotation:(id)annotation
{
    if ([MKDirectionsRequest isDirectionsRequestWithURL:url]) {
        MKDirectionsRequest *request = [[MKDirectionsRequest alloc]
                                       initWithContentsOfURL:url];
        MKMapItem *startItem = [request source];
        MKMapItem *endItem = [request destination];
        // YOUR CODE HERE
    }
}
```

Demo

Using the MKDirectionsRequest to provide a route

Making Great Routing Apps

Best practices

Making Great Routing Apps

Best practices

- Minimize user input after launch

Making Great Routing Apps

Best practices

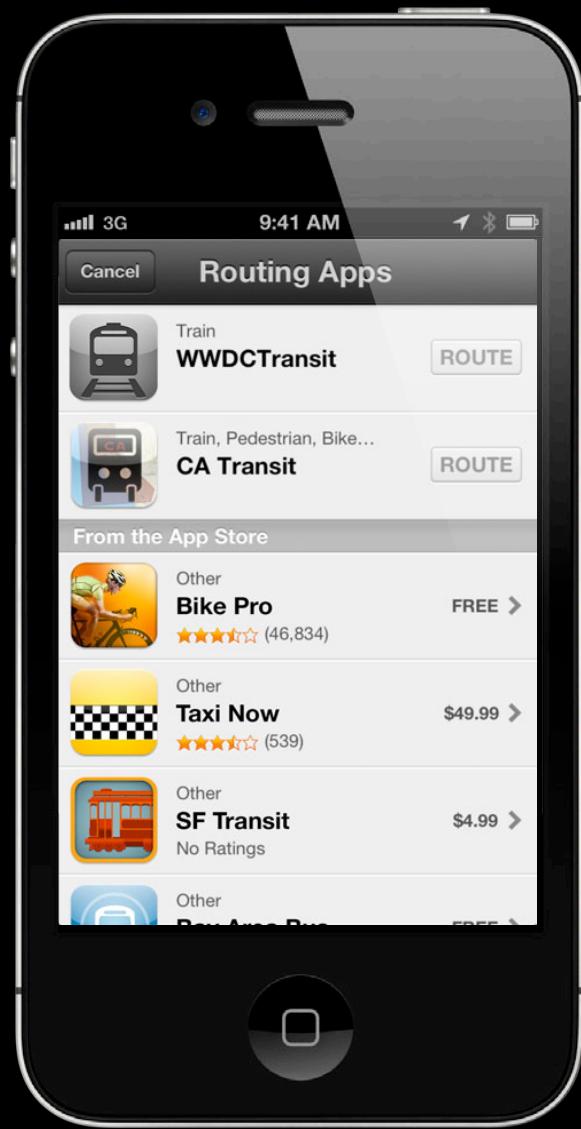
- Minimize user input after launch
- Personalize display using MKMapItem data

Making Great Routing Apps

Best practices

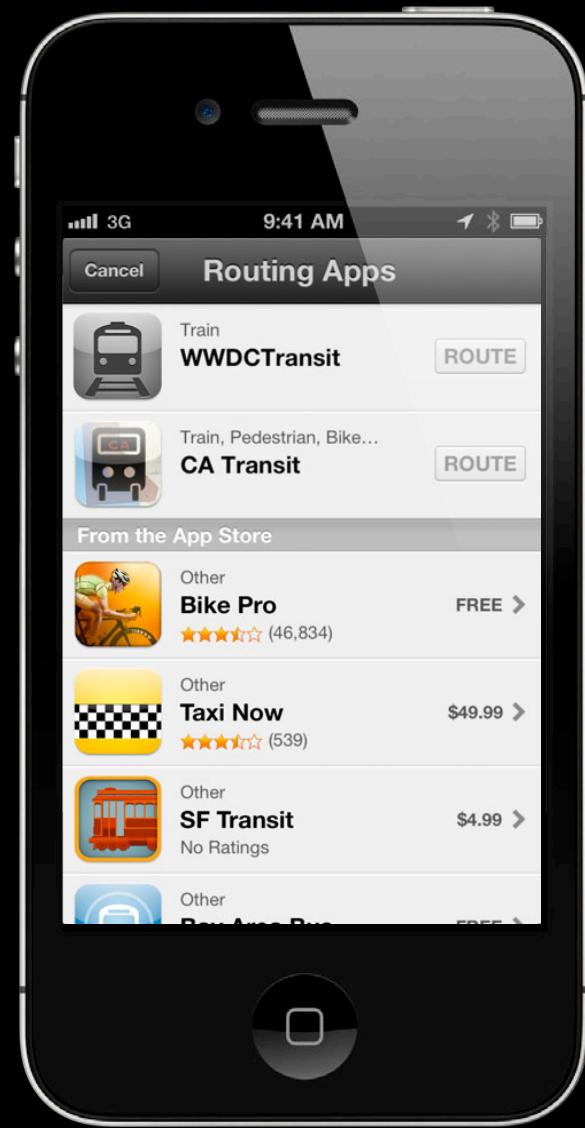
- Minimize user input after launch
- Personalize display using MKMapItem data
- Be reasonable with geographic coverage

Routing Apps



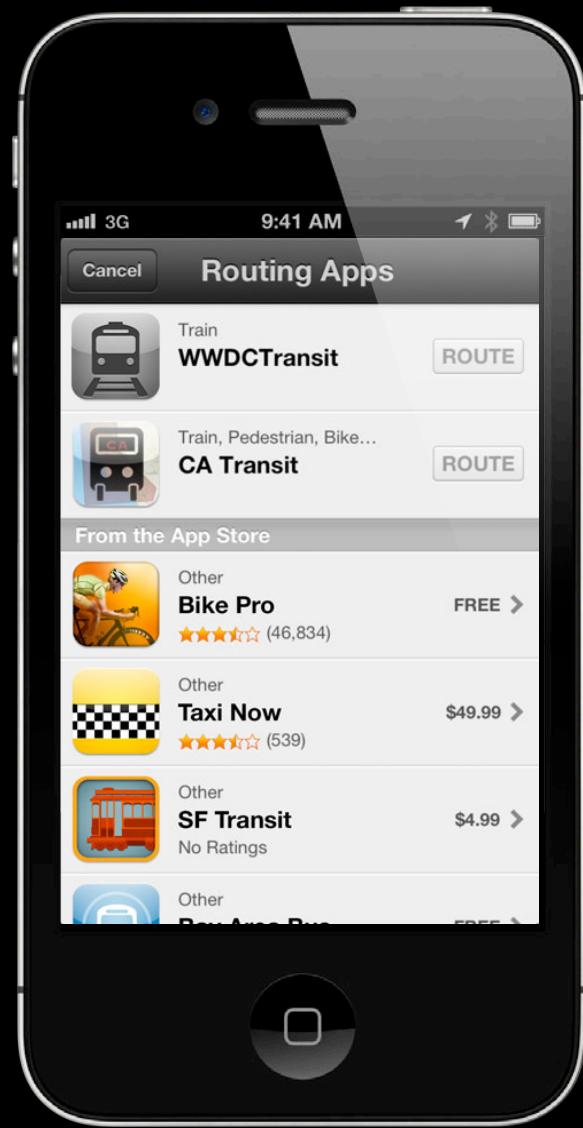
Routing Apps

- Integrate with Maps



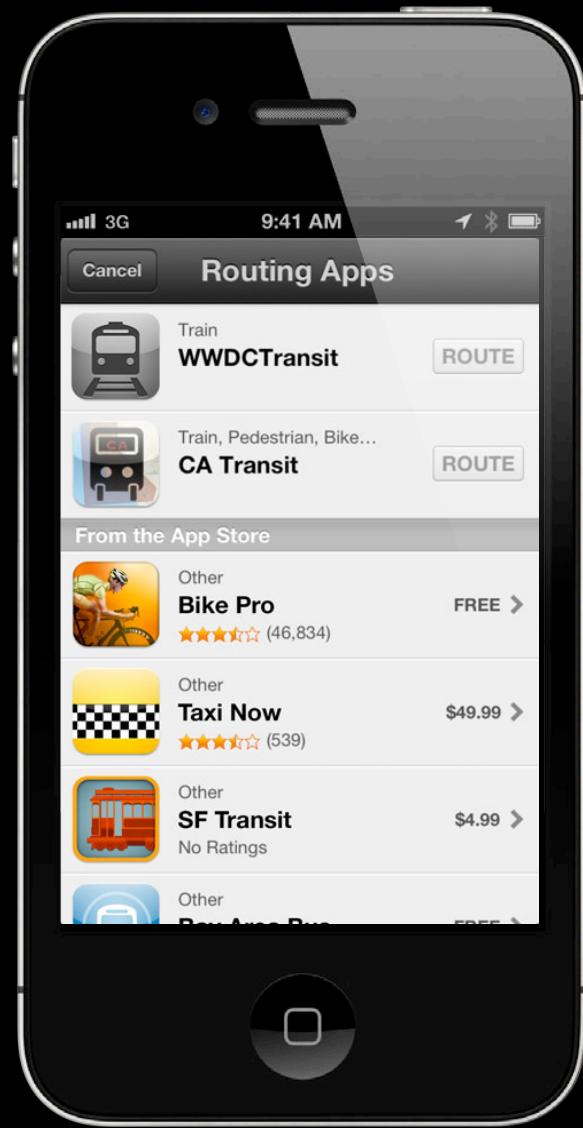
Routing Apps

- Integrate with Maps
- Increase traffic to your app



Routing Apps

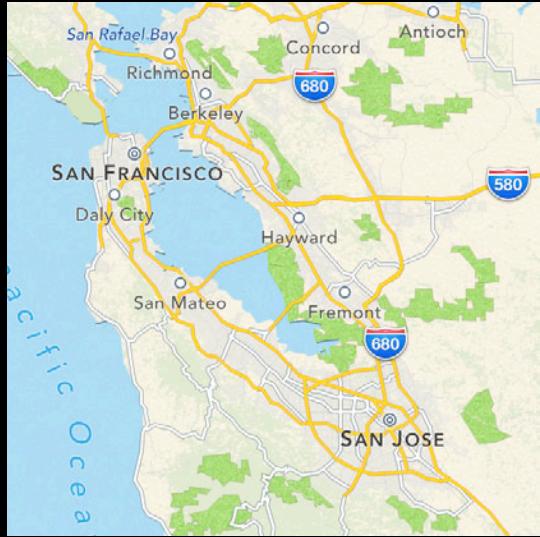
- Integrate with Maps
- Increase traffic to your app
- Get promoted to users who care



Summary

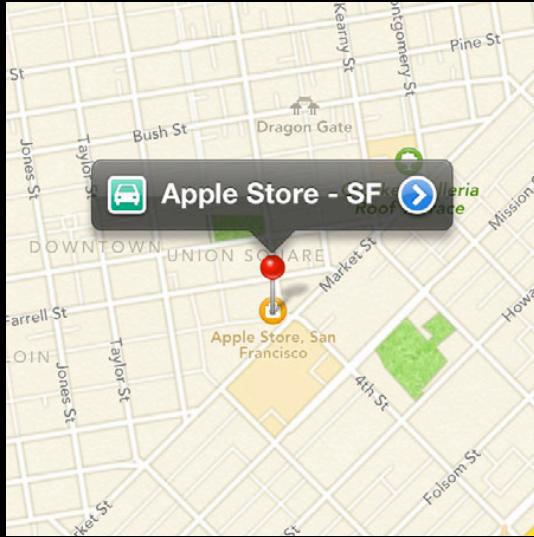


Summary



New Maps. Same API.

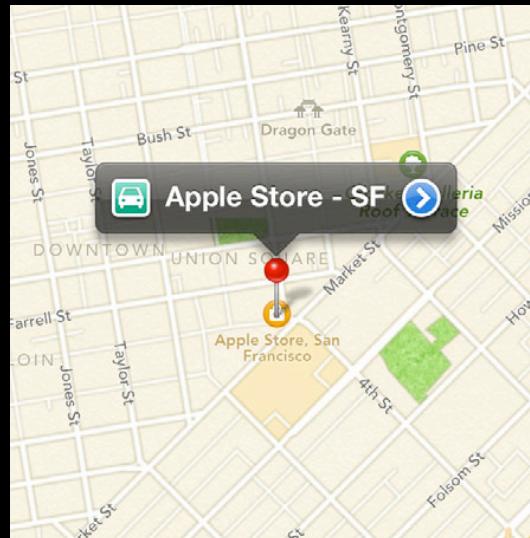
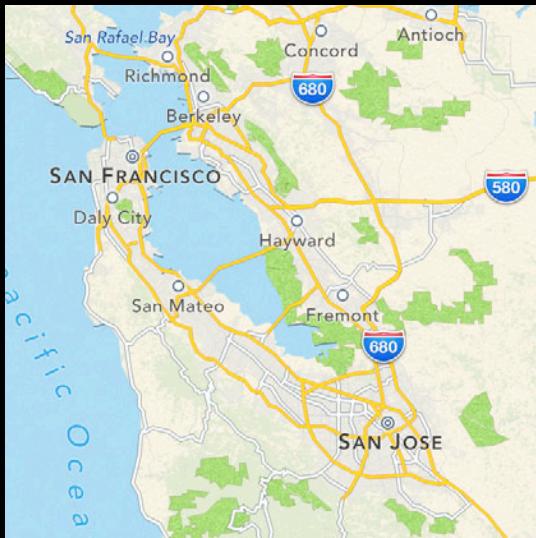
Summary



New Maps. Same API.

Open Maps with
MKMapItem

Summary



New Maps. Same API.

Open Maps with
MKMapItem

Routing Apps

More Information

Paul Marcos

Application Services Evangelist

pmarcos@apple.com

Documentation

MKMapView

http://developer.apple.com/library/ios/#documentation/MapKit/Reference/MKMapView_Class/

Location Awareness Programming Guide

<http://developer.apple.com/library/ios/#DOCUMENTATION/UserExperience/Conceptual/LocationAwarenessPG/>

WWDC 2011

“Visualizing Information Geographically with MapKit”

Apple Developer Forums

<http://devforums.apple.com>

Related Sessions

Getting Around with Map Kit (Repeat)

Pacific Heights
Tuesday 3:15PM

What's new in iTunes Connect for App Developers

Nob Hill
Thursday 9:00AM

Staying on Track with Location Services

Nob Hill
Wednesday 2:00PM

Labs

MapKit Lab #1

App Services Lab B
Tuesday 4:30PM

MapKit Lab #2

App Services Lab B
Thursday 11:30AM

