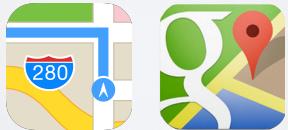


# Working with the iOS Core Location Framework and Google Maps SDK

## Lecture 12

Jonathan R. Engelsma



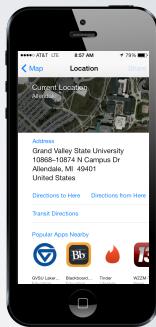
## TOPICS

- The Fundamentals.
- Working with CoreLocation
- Working with Google Maps SDK



## THE CURRENT LOCATION

- The ability to obtain the device's current location is a very powerful feature!
- Enabler for context aware computing
- App integrates content and behavior relevant to current location



## OBTAINING THE LOCATION

- Smartphones use a variety of techniques to locate the device:
  - GPS
  - Assisted GPS
  - Cellular / WiFi Triangulation



## GLOBAL POSITIONING SYSTEM

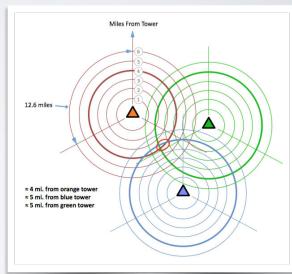
- Space-based satellite system (US Government)
- Requires line of sight to 4+ satellites
- Accuracy: ~ 3 - 15 meters.
- Can take time to get first fix.
- Don't work well indoors.

## ASSISTED GPS

- Greatly reduces the TTFF (time to first fix).
- Uses terrestrial radio network to accelerate connection to satellite.
- Widely used by modern cell phones.

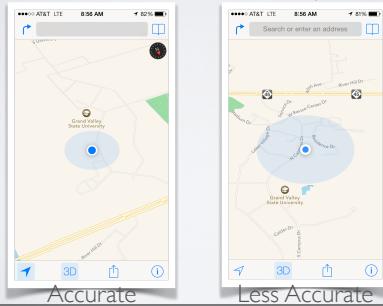
## WIFI / CELLULAR TRIANGULATION

- Uses terrestrial cellular towers and wifi access points to determine device location.
- Can be used indoors!



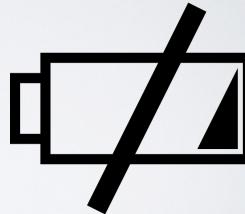
## OBTAINING THE LOCATION

- Implication: Location data varies in accuracy.



## TAKES ENERGY!

- Getting a GEO fix takes a lot of battery!
  - device can't sleep when GPS receiver is running.
  - make sure your apps don't misuse or you will get uninstalled real fast!



## CORE LOCATION

- CoreLocation Framework
  - Provides apps with location coordinate and heading info.
  - Can define geographical regions and monitor when user crosses region boundaries.
  - Also supports "beacon" regions with iBeacon devices.

## LOCATION MANAGER

- CLLocationManager:

```
var locationManager = CLLocationManager()  
locationManager.desiredAccuracy = kCLLocationAccuracyBest  
locationManager.distanceFilter = 10.0
```



Determine how far the device must travel before a new location update is generated.

## LOCATION ACCURACY

Accuracy	Technology
kCLLocationAccuracyBestForNavigation	GPS
kCLLocationAccuracyBest	GPS
kCLLocationAccuracyNearestTenMeters	GPS
kCLLocationAccuracyHundredMeters	WiFi (GPS in rural areas)
kCLLocationAccuracyKilometer	Cell tower
kCLLocationAccuracyThreeKilometers	Cell tower

## LOCATION MANAGER

- CLLocationManager requires location updates asynchronously.
- View Controller typically implements the CLLocationManagerDelegate protocol to receive these updates.

## LOCATION MANAGER DELEGATE

```
func locationManager(manager: CLLocationManager!,  
didChangeAuthorizationStatus status: CLAuthorizationStatus)  
{  
    // called when authorization changes.  
}  
  
func locationManagerDidResumeLocationUpdates(manager: CLLocationManager!) {  
    // called when location updates start to arrive.  
}
```

## LOCATION MANAGER DELEGATE

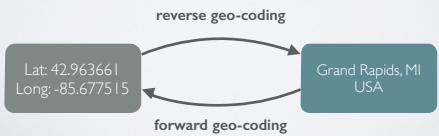
```
func locationManager(manager: CLLocationManager!,  
didUpdateLocations locations: [AnyObject]!)  
{  
    var location: CLLocation = locations.first as! CLLocation  
    // put code to handle location update here.  
}
```

## CLLOCATION

- CLLocation: represents the data generated by CLLocationManager. It contains:
  - geo coordinates and altitude
  - accuracy indicator
  - time
  - speed and heading.

## GEO CODING

- CLGeocoder: converts between a lat/long coordinate and a user-friendly representation of that coordinate.



## GEO-CODING GUIDELINES

- Geocoder requests involve the network and are rate limited:
  - send one request per user action.
  - cache the results if you do more than one action per loc.
  - issue a new geocoding request only if user has moved a significant distance.
  - don't use it in the background!

## GOOGLE MAPS SDK

- Google Maps SDK is the alternative of choice for Apple MapKit!
- Provides more detailed map tiles.
- Includes Street View!
- Can optionally use Apple's MapKit framework



<https://developers.google.com/maps/documentation/ios/intro>

## GOOGLE MAPS SDK

- Types: Normal, Satellite, Hybrid, Terrain
- Traffic Layer
- Indoor building maps where available. (Museum navigator!)
- Places API / Place Picker
- Panorama (StreetView)
- Interactive Markers
- Polygons
- Camera view animation



## DISPLAY MODES



Standard



Satellite



StreetView

## CREATING A MAPVIEW

```
override func viewDidLoad() {
    super.viewDidLoad()

    locationManager.delegate = self
    locationManager.requestWhenInUseAuthorization()
    var cameraAllendale = GMSCameraPosition.cameraWithLatitude(42.96356,
        longitude: -85.8899, zoom: 14.7)

    mapView = GMSMapView.mapWithFrame(CGRectZero, camera: cameraAllendale)
    mapView.settings.compassButton = true
    self.view = mapView

    let marker = GMSMarker()
    marker.position = CLLocationCoordinate2DMake(42.96356, -85.8899)
    marker.title = "GVSU"
    marker.snippet = "Allendale, MI"
    marker.icon = UIImage(named: "gvsu")
    marker.appearAnimation = kGMSMarkerAnimationPop
    marker.map = mapView
}
```

## READING ASSIGNMENT

- Chapter 21 in Programming iOS8 by Neuburg
- Google Map SDK Documentation / Example Code
  - <https://developers.google.com/maps/documentation/ios/>