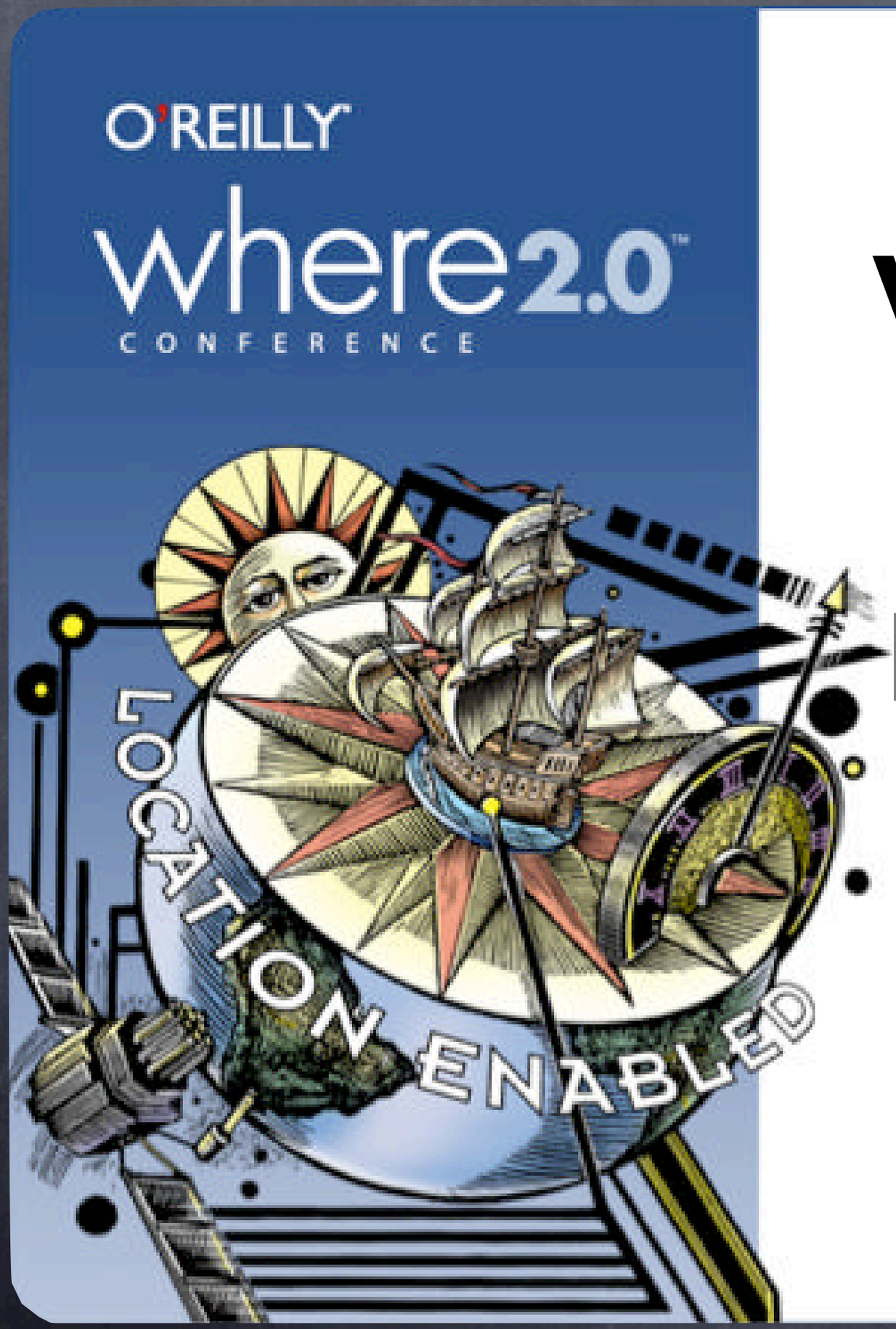



# To boldly go where no man has gone before:

## Exploring Geo in iPhone & Android

Apr 1, 2010 Thu
2:50 PM San Jose


# To boldly go where no man has gone before:

## Exploring Geo in iPhone & Android

Stardate: -312751.7313546423

# Three Years Ago

## Jan 2007

Stardate: -315974.3150684931

# Widescreen iPod with touch controls

## Stardate: -315974.31506849 31

# Revolutionary mobile phone

## Stardate: -315974.3150684931


# Breakthrough internet communicator

## Stardate: -315974.3150684931

iPod

Phone


Internet

Stardate: -315974.3150684931

# iPhone

Stardate: -315974.3150684931

# Chief Science Officer



**Stardate: -314000**



**Portable Sensing**

**Data Analysis**

**Data Communications**

**Stardate: -314000**

# Tricorder

Stardate: -314000


# iPhone

**Stardate: -315974.3150684931**


Tricorder
Stardate: -314000

Tricorder


iPhone

Stardate: -314000


Locator

Scanner


Compass

Recorder

Sensor

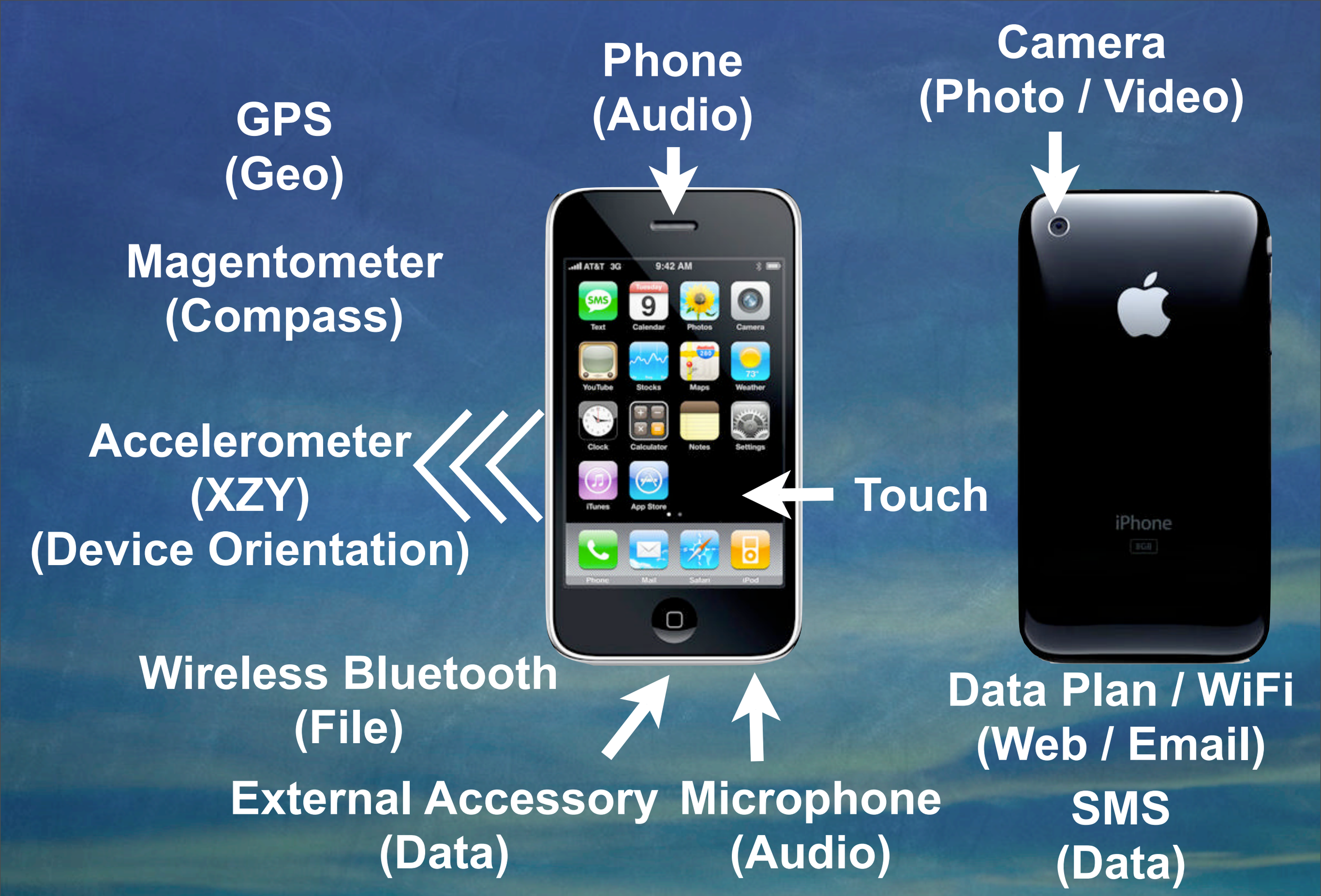

Search


Text

Data

Voice

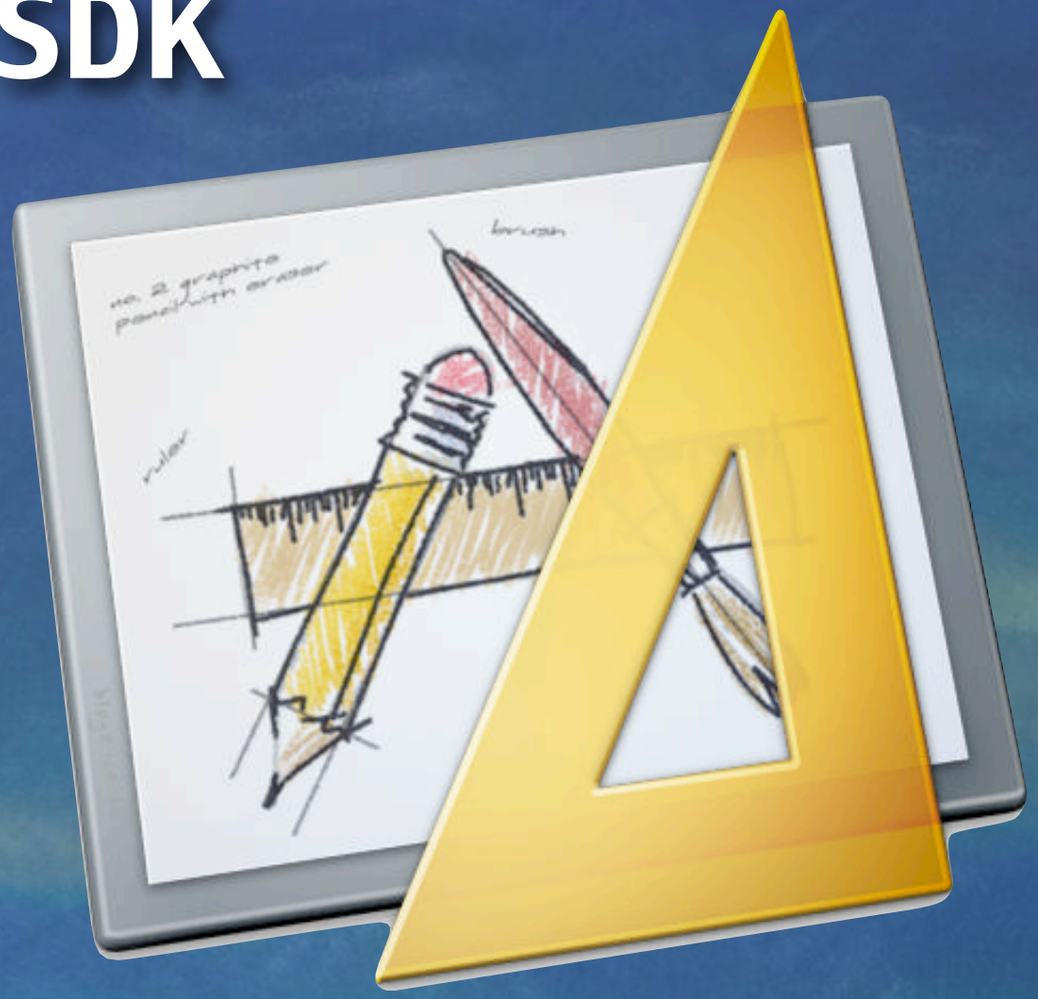
Stardate: -314000




# iPhone's Anatomy


# Portable Sensors

# iPhone SDK

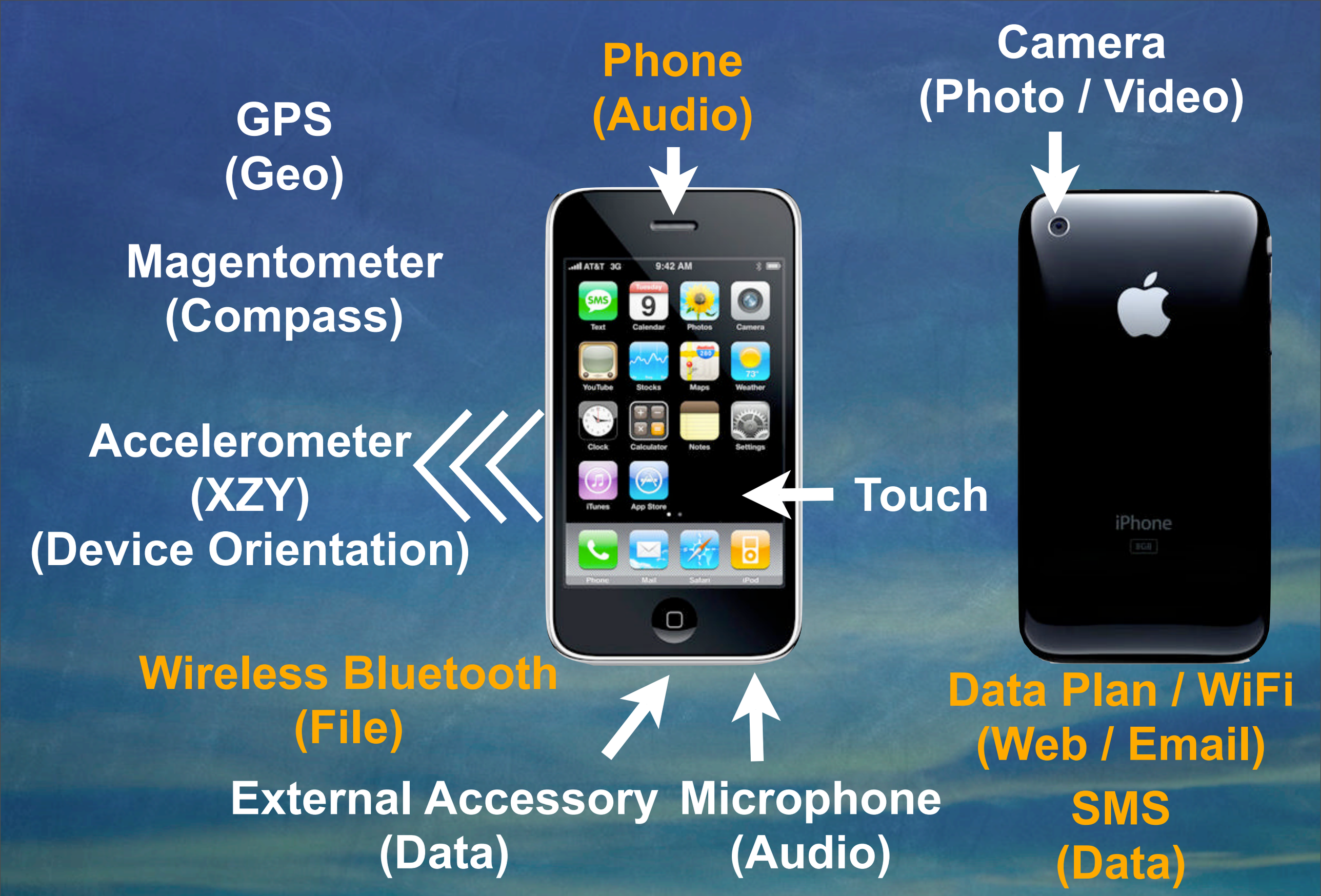




Xcode

Interface Builder



Data Analysis




# Data Communications

Stardate: -315974.3150684931


MapKit


Core Location

# iPhone SDK Frameworks


# iPhone SDK Frameworks

Cocoa Touch Layer

Media Layer

Core Services Layer

Core OS Layer


# iPhone SDK Frameworks

## Cocoa Touch Layer
Map Kit

## Media Layer

## Core Services Layer
Core Location

## Core OS Layer

iPhone SDK Frameworks


# CLLocation Manager
# CLLocation
# CLHeading

## Core Location Framework: Class

# UIViewController Interface

## CLLocationManager

```
#import <CoreLocation/CoreLocation.h>
@interface GetLocationViewController : UIViewController
<CLLocationManagerDelegate> {
    CLLocationManager *locationManager;
    CLLocation *bestEffortAtLocation;
}
@property (nonatomic, retain) CLLocationManager
*locationManager;
@property (nonatomic, retain) CLLocation
*bestEffortAtLocation;
```

# ViewController Method

## CLLocationManager

```
// Create the manager object
self.locationManager = [[[CLLocationManager alloc] init]
autorelease];
locationManager.delegate = self;
locationManager.desiredAccuracy = [[setupInfo
objectForKey:kSetupInfoKeyAccuracy] doubleValue];
[locationManager startUpdatingLocation];
```


# CLLocation Class Constants

- CLLocationDegrees
- CLLocationCoordinate2D
- CCLocationAccuracy
- Accuracy Constants
- CLLocationSpeed
- CLLocationDirection

# Accuracy Constants

## CLLocation Class

**locationManager.desiredAccuracy** is the most important property of Location Manager. It determines the **amount of power** it consumed.

**Constant values are to specify the accuracy of a location.**

| | |
|---|---|
| Best | kCLLocationAccuracyBest |
| 10 Meters | kCLLocationAccuracyNearestTenMeters |
| 100 Meters | kCLLocationAccuracyHundredMeters |
| 1000 Meters | kCLLocationAccuracyKilometer |
| 3000 Meters | kCLLocationAccuracyThreeKilometers |

# CLLocation Class Constants


## CLLocationDegrees

Delivers a latitude or longitude value specified in **degrees**. Data type is **double**.

## CLLocationSpeed

Delivers the speed at which the device is moving in **meters per second**. Data type is **double**.

# CLLocation Class Constants

## CLLocationDirection

Delivers a direction that is measured in **degrees** and relative to true north. Data type is **double**.

- North is 0 degrees
- East is 90 degrees
- South is 180 degrees
- Any "-" value indicates an invalid direction

# Clockwise


# CCLocationDirection


# CLLocation Class Properties

- altitude
- coordinate
- course
- horizontalAccuracy
- speed
- timestamp
- verticalAccuracy


# Read-Only

- altitude
- coordinate
- course
- horizontalAccuracy
- speed
- timestamp
- verticalAccuracy


# Measurement Units

- altitude (meters)
- coordinate
- course (degrees)
- horizontalAccuracy (meters)
- speed (meters per sec)
- timestamp (NSDate)
- verticalAccuracy (meters)


# CLLocation Manager

## CLLocation

## CLHeading


# CLLocationManager

## Core Location

Create a CLLocationManager object to get heading by invoking [CLLocationManager startUpdatingHeading].

iPhone 3GS contains a magnetometer – a magnetic field detector. It displays the raw x, y, and z magnetometer values. Magnitude of the magnetic field is computed in strength.


# CLLocationManager Class

## Properties

- headingAvailable
- headingFilter

## Instance Methods

- startUpdatingHeading
- stopUpdatingHeading

## Constants

- Heading Filter Value

# CLLocationManager

## Core Location

```
if (locationManager.headingAvailable == NO) {
        self.locationManager = nil; // No compass is available}
        else {
       // heading service configuration
       locationManager.headingFilter = kCLHeadingFilterNone;
       // setup delegate callbacks
       locationManager.delegate = self;
       // start the compass
       [locationManager startUpdatingHeading];
    }
}
```


# CLLocation Manager

# CLLocation

# CLHeading

## Core Location Framework: Class

# CLHeading

## Core Location

```
- (void)locationManager:(CLLocationManager *)manager
didUpdateHeading:(CLHeading *)heading {
  // Update the labels with the raw x, y, and z values.
  [xLabel setText:[NSString stringWithFormat:@"%.1f", heading.x]];
  [yLabel setText:[NSString stringWithFormat:@"%.1f", heading.y]];
  [zLabel setText:[NSString stringWithFormat:@"%.1f", heading.z]];
}
```

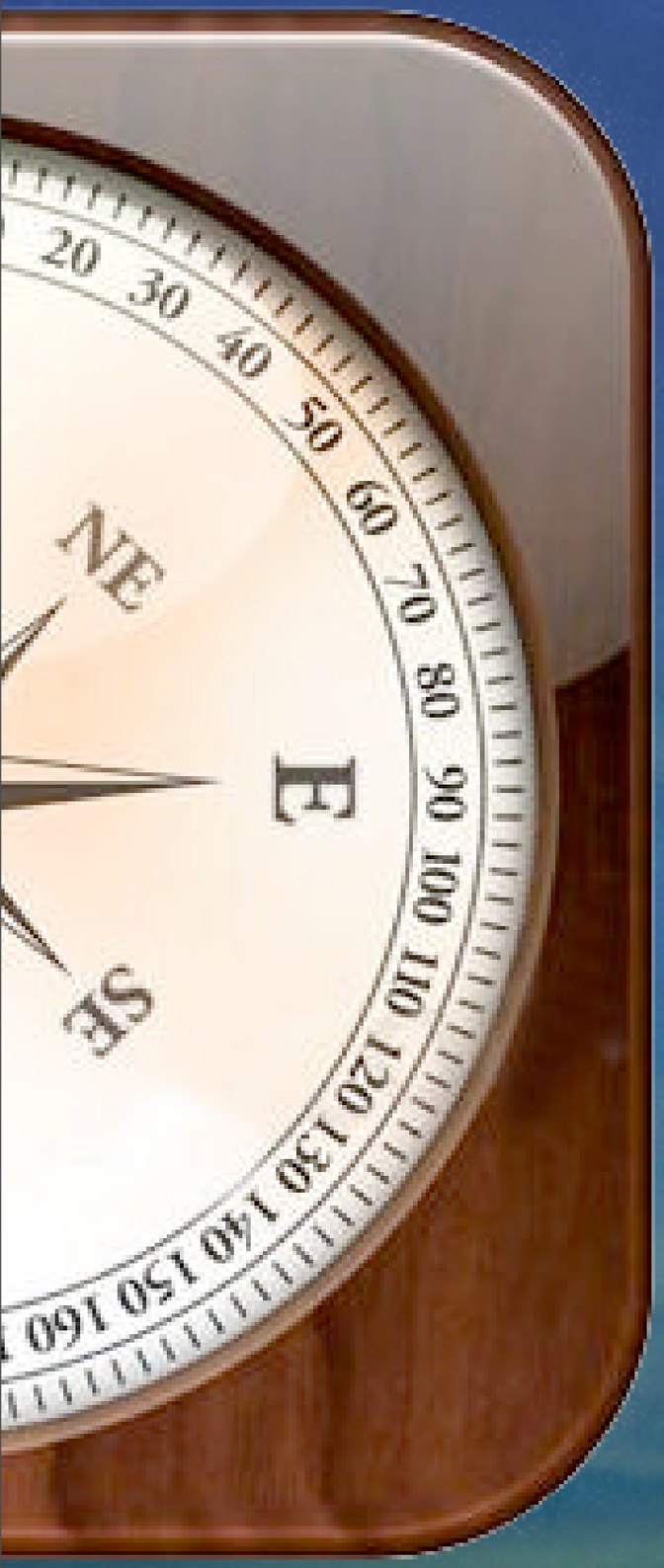

MapKit


Core Location

# iPhone SDK Frameworks

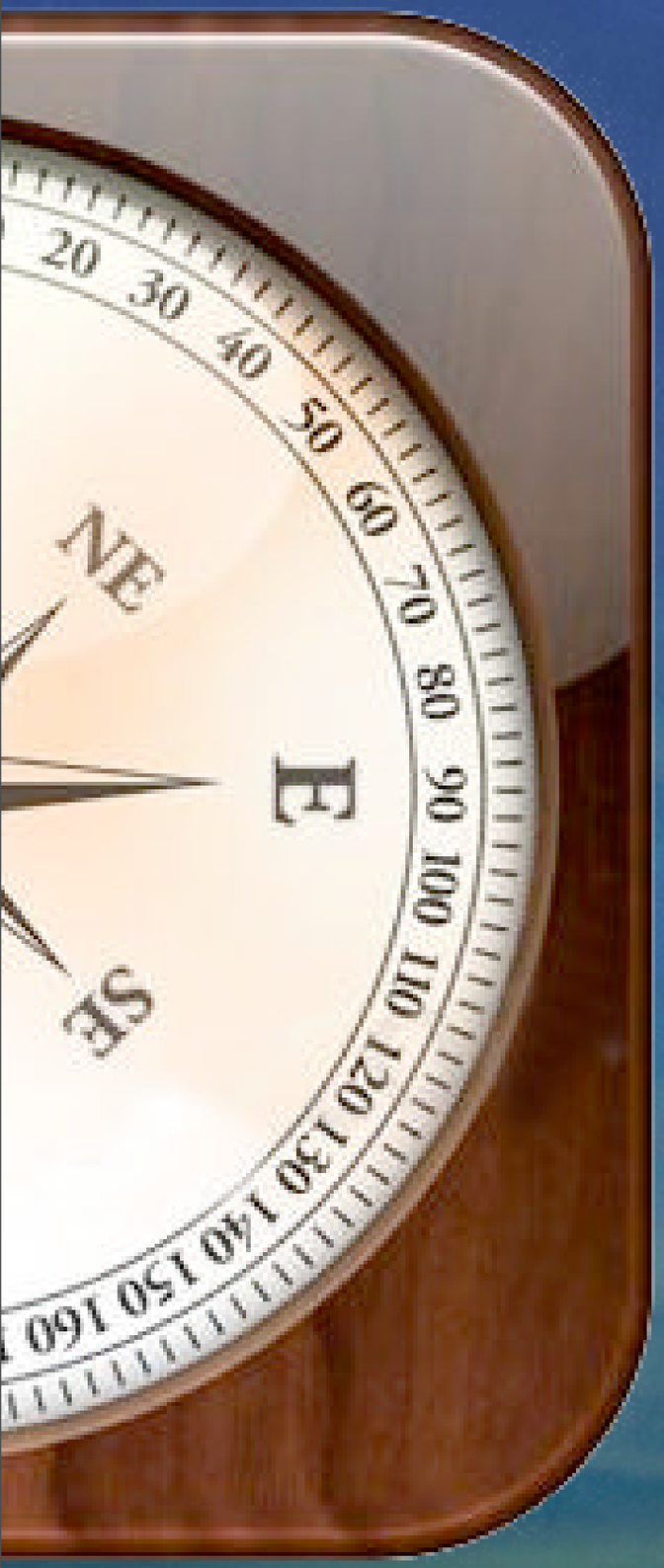

- MKAnnotationView
- MKMapView
- MKPinAnnotationView
- MKPlacemark
- MKReverseGeocoder
- MKUserLocation

# MapKit Framework: Class




# MKReverseGeocoder

MKReverseGeocoder offers services to convert a map coordinate (latitude & Longitude) to info such as country, city, or street. It works with a network-based map service to look up placemark information for a specified coordinate value.




# MKReverseGeocoder

- Each app is limited to amount of reverse geocoding
- Send one reverse-geocoding request for any one user action
- Reuse the results from initial request
- Suggest not to send one reverse-geocode request per minute


- MKAnnotationView
- MKMapView
- MKPinAnnotationView
- MKPlacemark
- MKReverseGeocoder
- MKUserLocation

# MapKit Framework: Class

# MKMapView Class Properties

- annotations
- annotationsVisibleRect
- centerCoordinate
- delegate
- mapType
- region
- scrollEnabled
- selectedAnnotations
- showsUserLocation
- userLocation
- userLocationVisible
- zoomEnabled

# MKMapView Class

## MKMapType

It delivers the type of map to display.

- MKMapTypeStandard
- MKMapTypeSatellite
- MKMapTypeHybrid


- MKAnnotationView
- MKMapView
- MKPinAnnotationView
- MKPlacemark
- MKReverseGeocoder
- MKUserLocation

# MapKit Framework: Class

# MKAnnotationView Class Properties

- annotation
- image
- calloutOffset
- leftCalloutAccessoryView
- canShowCallout
- reuseIdentifier
- centerOffset
- rightCalloutAccessoryView
- enabled
- selected


- MKAnnotationView
- MKMapView
- MKPinAnnotationView
- MKPlacemark
- MKReverseGeocoder
- MKUserLocation

# MapKit Framework: Class

# MKAnnotationView Class Properties


## Properties

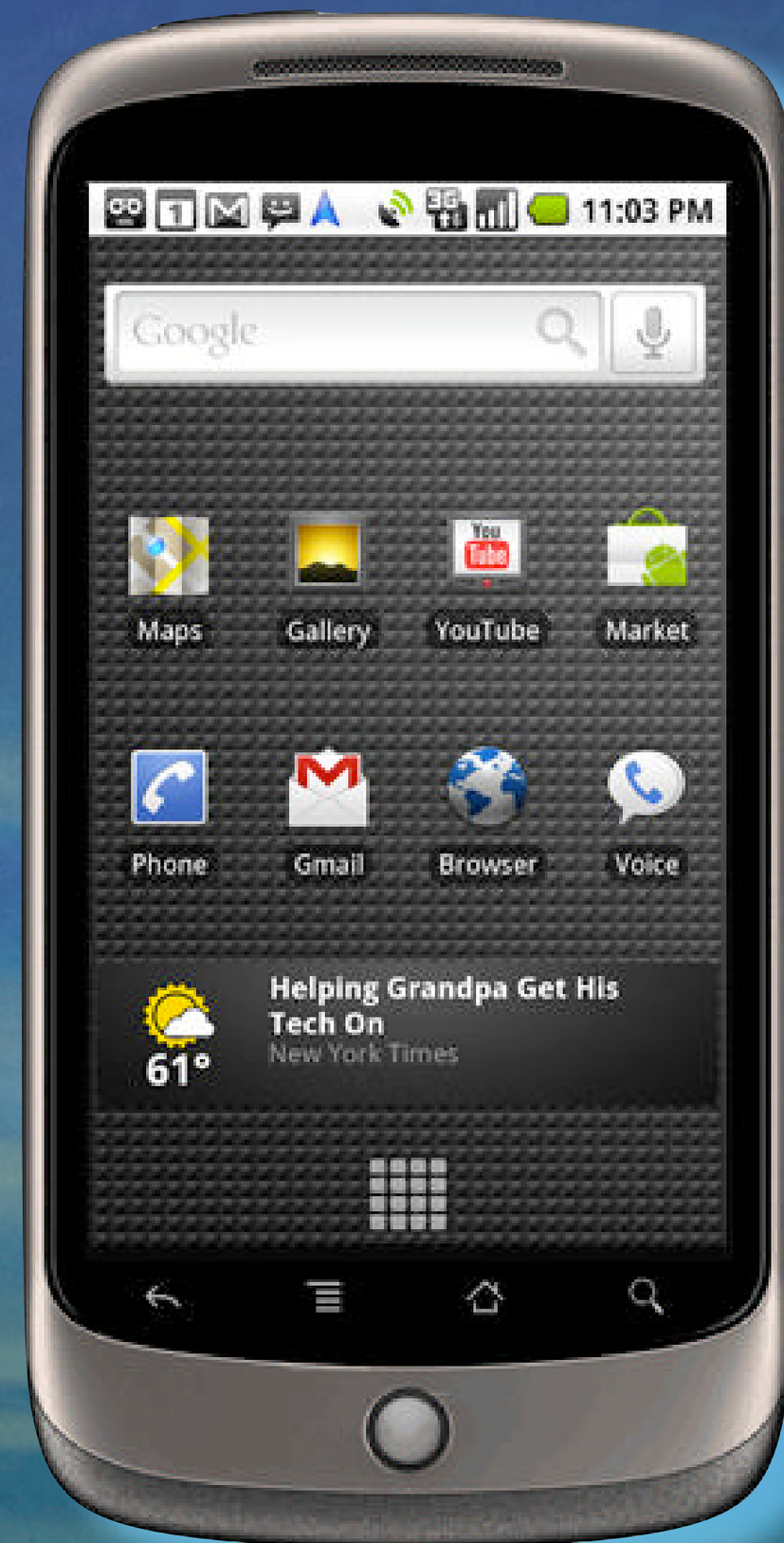
**animatesDrop**

**pinColor**

## Constants

**MKPinAnnotationColor**

- MKPinAnnotationColorRed (Destination Points)
- MKPinAnnotationColorGreen (Starting Points)
- MKPinAnnotationColorPurple (User-specified Points)


# iPhone VS Android

# Google Maps External Library

# Location Services

## Android SDK

# Classes

- Address
- Criteria
- Geocoder
- GpsSatellite
- GpsStatus
- Location
- LocationManager
- LocationProvider

package: android.location

# Location Class Methods

- getAccuracy()
- getAltitude()
- getBearing()
- getExtras()
- getLatitude()
- getLongitude()
- getProvider()
- getSpeed()
- getTime()

package: android.location

# Google Maps External Library

# Location Services

## Android SDK

# Google Maps External Library

- Use Google APIs add-on
- Download Maps external library
- Must register with Google Maps service
- Obtain a Maps API Key

# AndroidManifest.xml

- Declare Maps Library
- Request internet permission
- Hide title bar

```
<uses-library
android:name="com.google.android.maps" />
<uses-permission
android:name="android.permission.INTERNET" />
<activity android:name=".HelloMaps"
android:label="@string/app_name"
android:theme="@android:style/Theme.NoTitleBar">
```

# res/layout/main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<com.google.android.maps.MapView
    xmlns:android="http://schmas.android.com/apk/res/android"
    android:id=@+id/mapview"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:clickable="true"
    android:apiKey="Map API Key"
/>
```

# HelloMaps.java

```
public class HelloMaps extends MapActivity
```

```
@Override protected boolean isRouteDisplayed() {
return false;
}
```

```
@Override public void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.main);
}
```

```
MapView mapView = (MapView)
findViewById(R.id.mapview);
mapView.setBuiltInZoomControls(true);
```



**PhoneGap**

**Titanium Mobile**

**3rd Party SDK: HTML / CSS / JavaScript**

```
function run() {
        var win = function(position) {                    // Grab
coordinates object from the Position object passed into success
callback.
            var coords = position.coords;
            // Call for static google maps data – make sure you use
your own Google Maps API key!
            var url = "http://maps.google.com/maps/api/staticmap?
center=" + coords.latitude + "," + coords.longitude +
"&zoom=13&size=320x480&maptype=roadmap&key=MyGoogleMapsA
PIKey&sensor=true";
                document.getElementById('map').setAttribute('src',url);
            };
            var fail = function(e) {
                alert('Can\'t retrieve position.\nError: ' + e);
            };
            navigator.geolocation.getCurrentPosition(win, fail);
        }
```

# 3rd Party SDK: PhoneGap



**PhoneGap**

**Titanium Mobile**

**3rd Party SDK: HTML / CSS / JavaScript**

# JavaScript Library

## geolocation.js

Corresponding to iphone SDK:
Core Location Framework: CLLocationManager & CLLocation

```
var longitude = e.coords.longitude;
var latitude = e.coords.latitude;
var altitude = e.coords.altitude;
var heading = e.coords.heading;
var accuracy = e.coords.accuracy;
var speed = e.coords.speed;
var timestamp = e.coords.timestamp;
var altitudeAccuracy = e.coords.altitudeAccuracy;
```

3rd Party SDK: Titanium

# JavaScript Library

## geolocation.js

Corresponding to iphone SDK:
Core Location Framework: CLHeading

```
var x = e.heading.x;
var y = e.heading.y;
var z = e.heading.z;
var magneticHeading = e.heading.magneticHeading;
var accuracy = e.heading.accuracy;
var trueHeading = e.heading.trueHeading;
var timestamp = e.heading.timestamp;
```

3rd Party SDK: Titanium

# JavaScript Library

## map_view.js

Corresponding to iphone SDK: MapKit Framework

```
var mapview = Titanium.Map.createView({
    mapType: Titanium.Map.STANDARD_TYPE,
    region: {latitude:33.74511, longitude:-84.38993,
latitudeDelta:0.01, longitudeDelta:0.01},
    animate:true,
    regionFit:true,
    userLocation:true,
    annotations:[apple, atlanta]
});
```

# Chief Science Officer



**Stardate: -314000**

# 01.12.2010

Earthdate

Stardate: -312969.29223744429


# Haiti 2010 Earthquake

**Stardate: -314000**

# 01.27.2010

Earthdate

Stardate: -312927.6255707762

# Medical Tricorder


STANFORD ENGINEERING

# STANFORD UNIVERSITY

## School of Engineering (EE46)

# Engineering For Good
# Save The World
## Have Fun Doing It

# STANFORD UNIVERSITY

# Save The World

- Malaria
- TB
- HIV

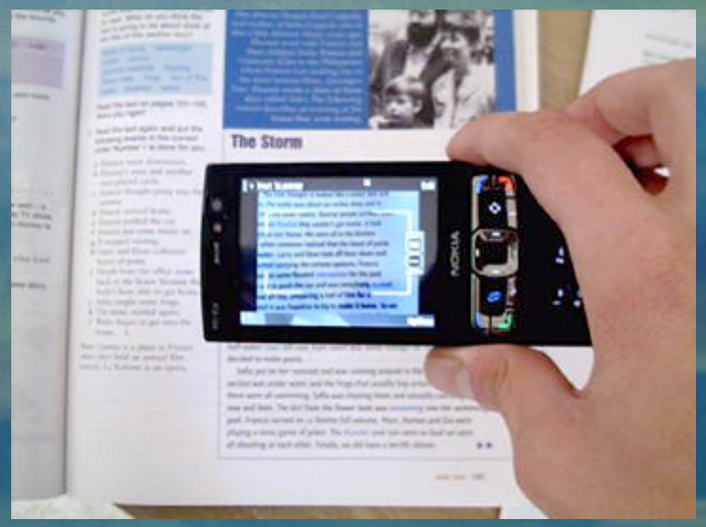
Camera
(Photo / Video)


Bar Code


QR Code

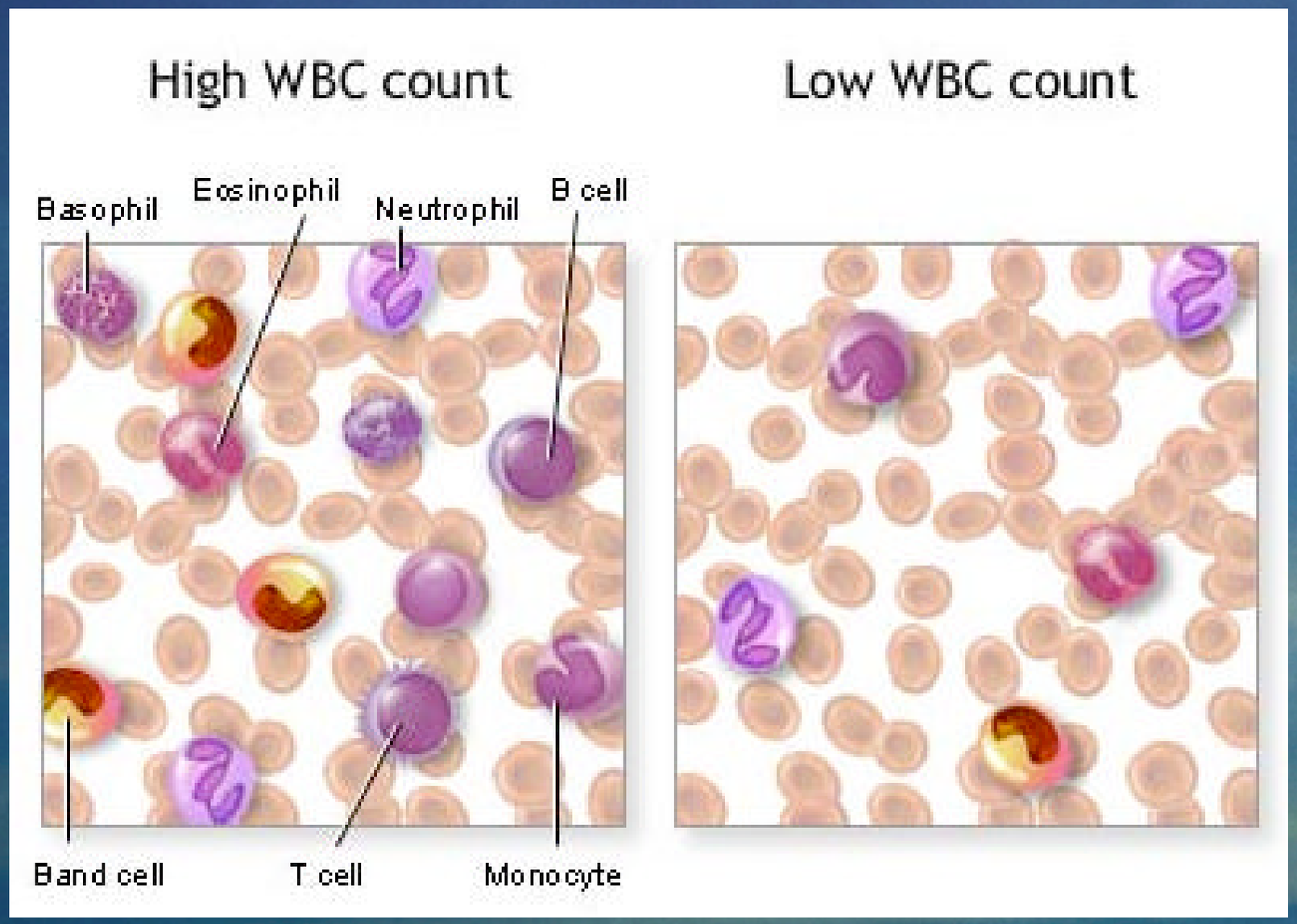
Augmented Reality (AR)


Optical Character Recognition (OCR)




Scanner: 2D Objects

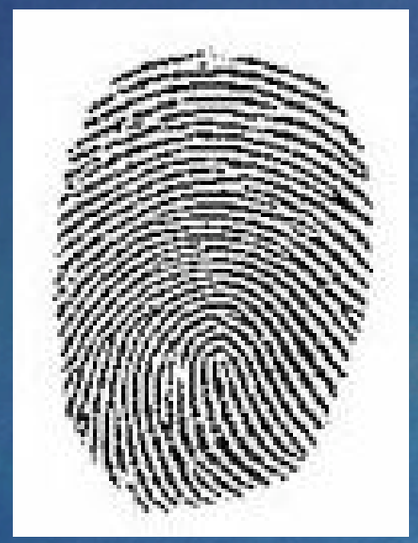
# Detect Diseases

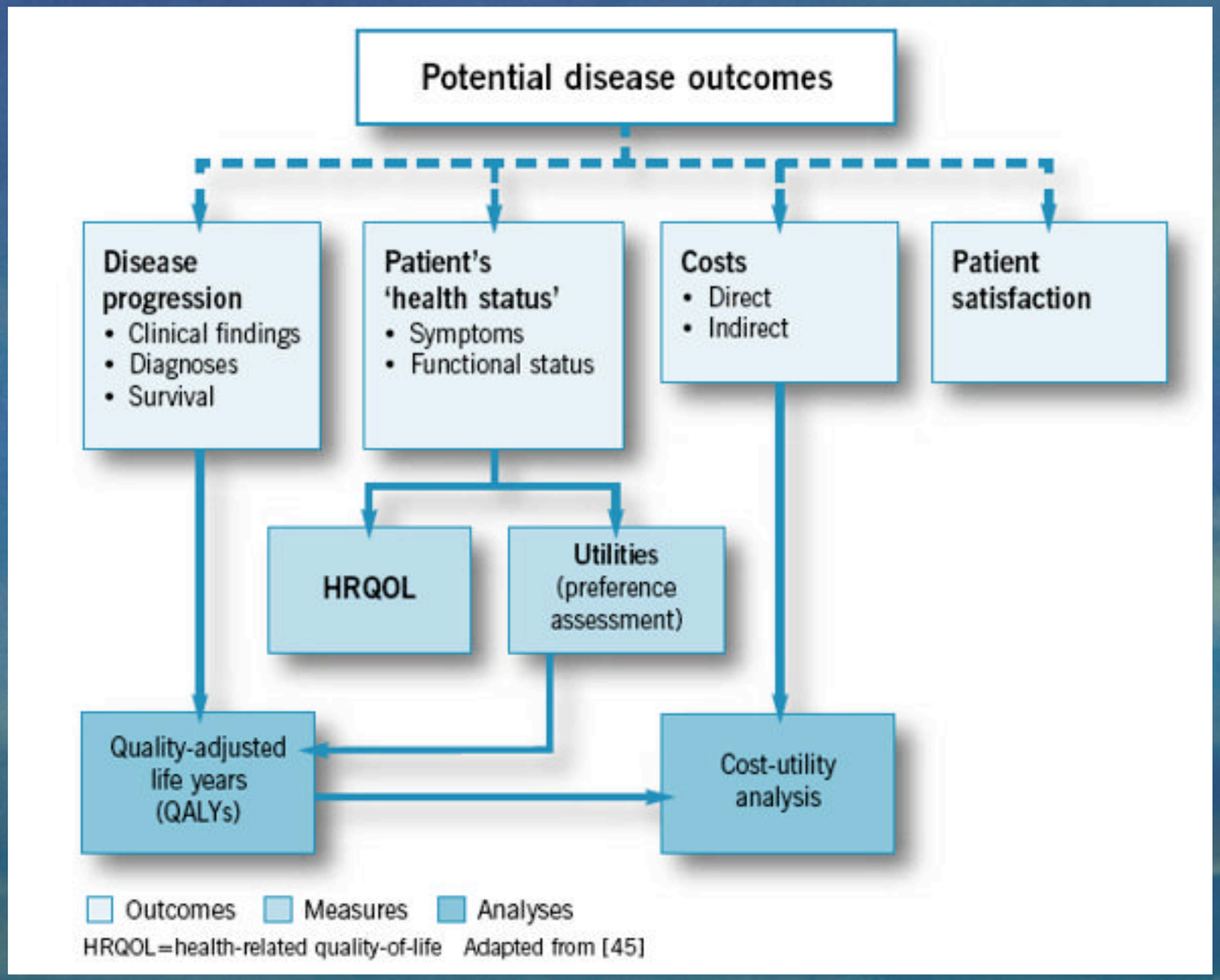




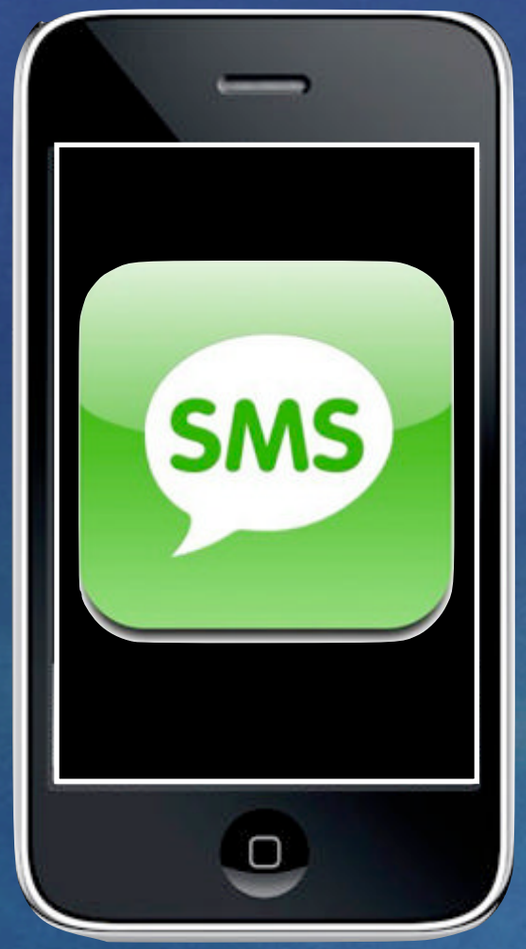
## Scanner: Life forms

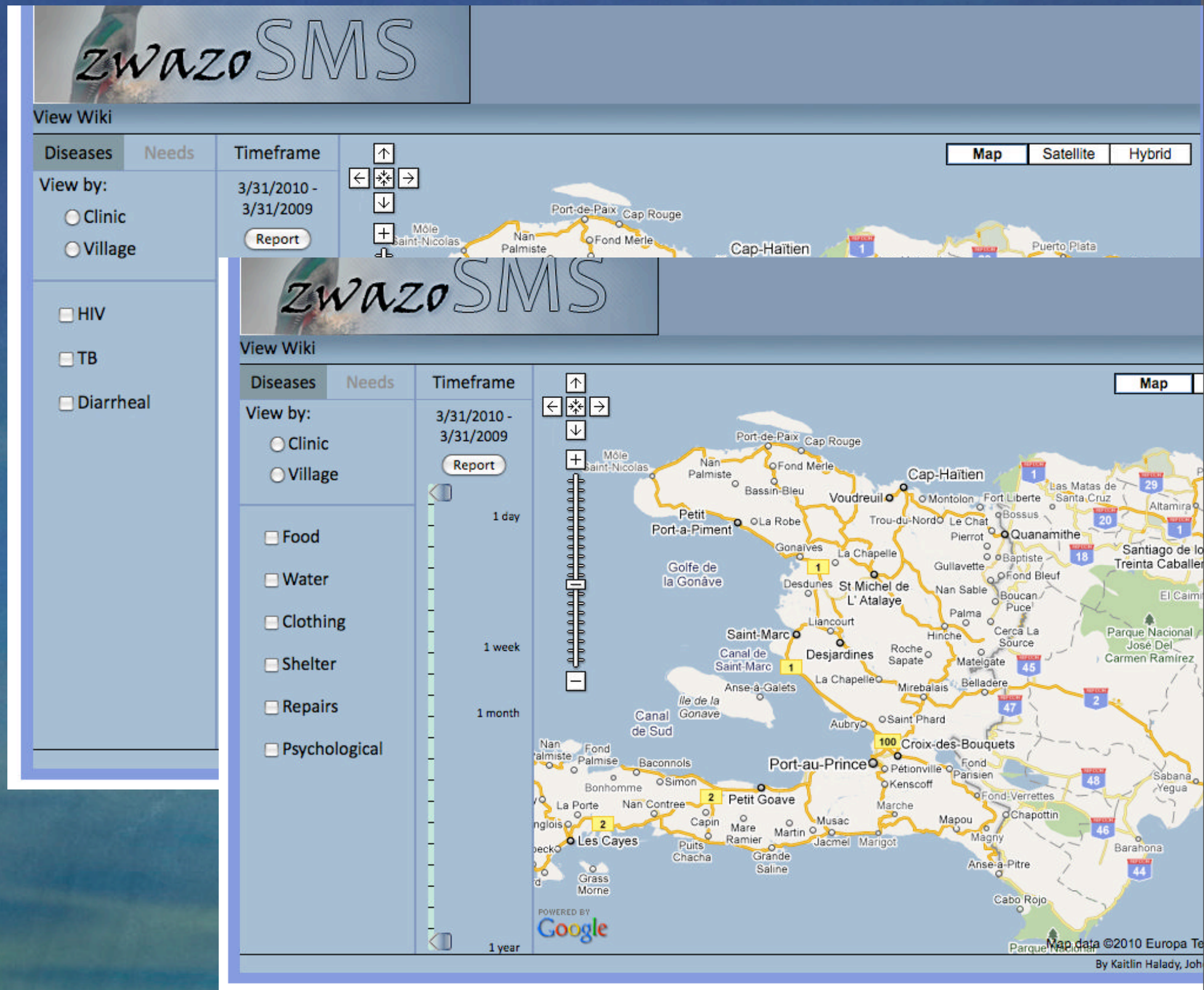
# Identify Patients



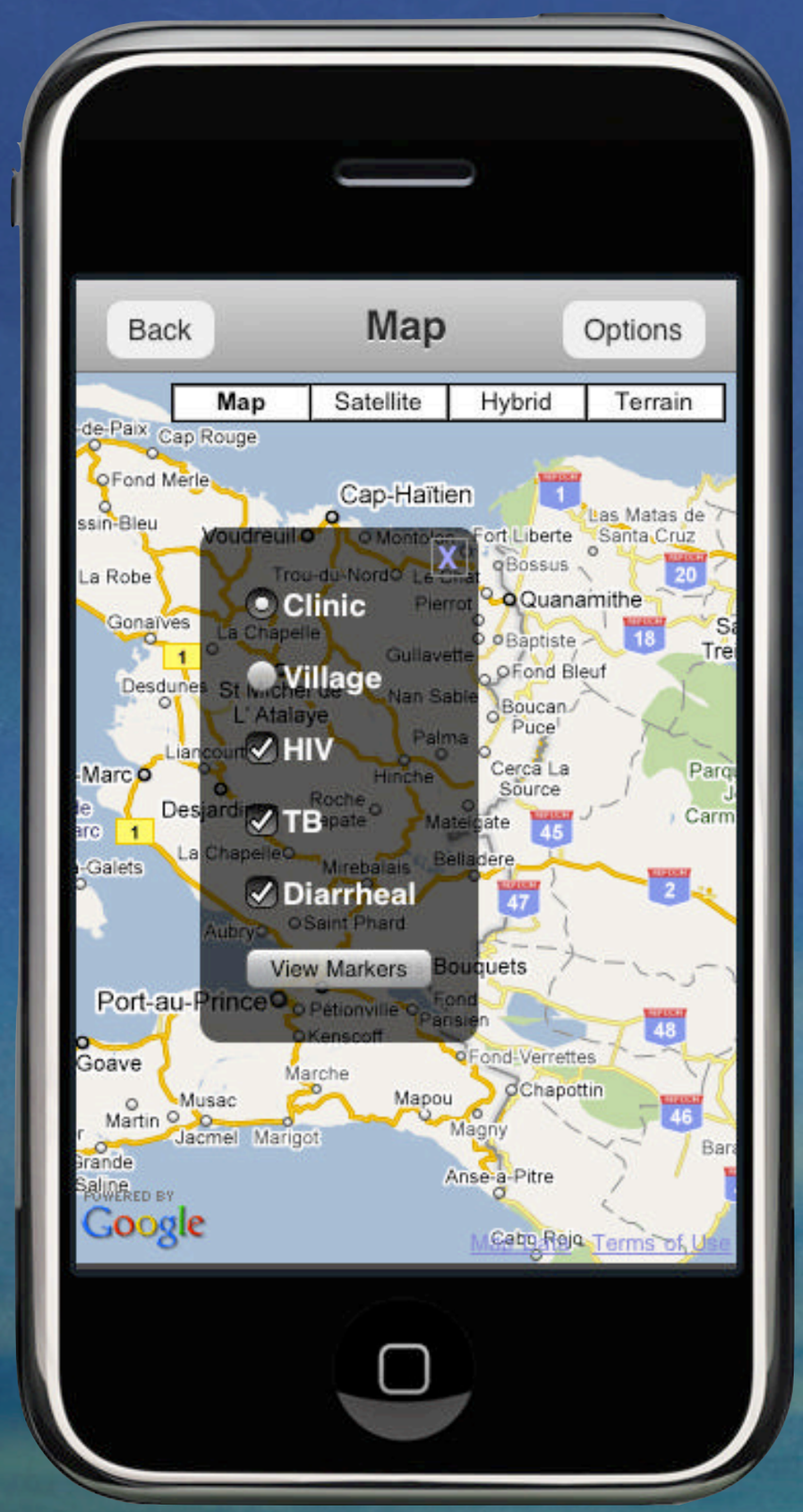


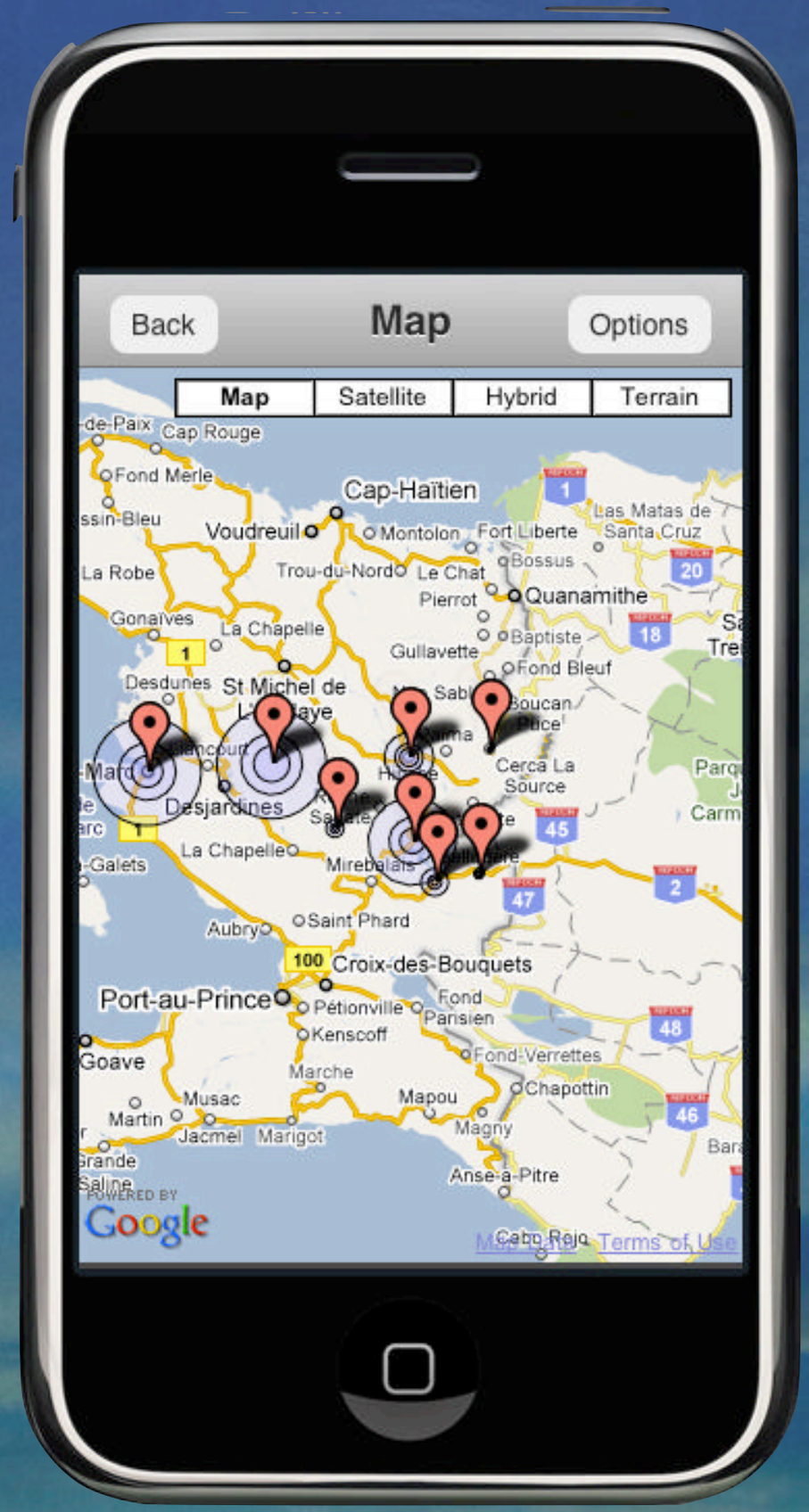

## Scanner: Life forms



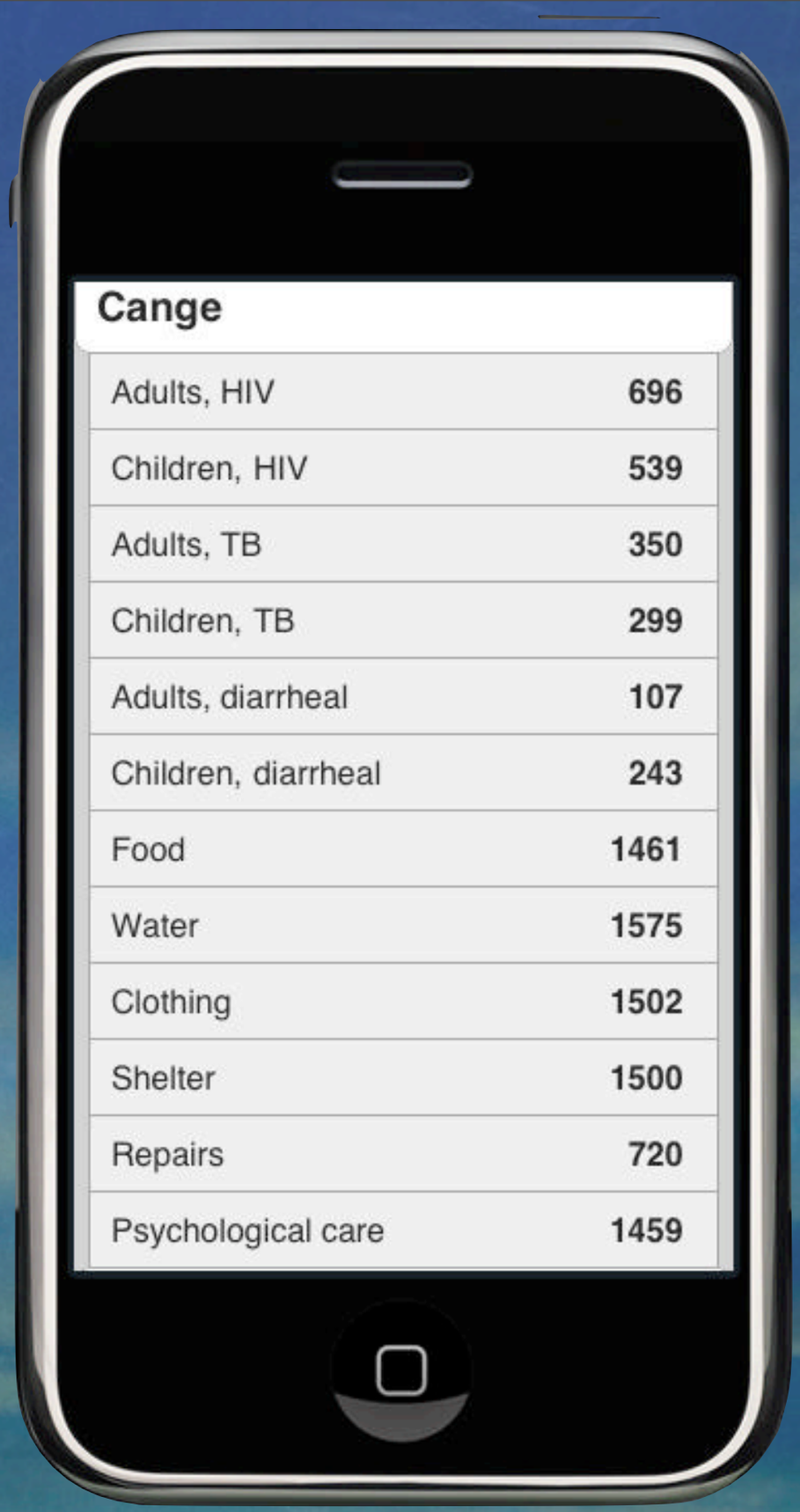




# Medical Tricorder



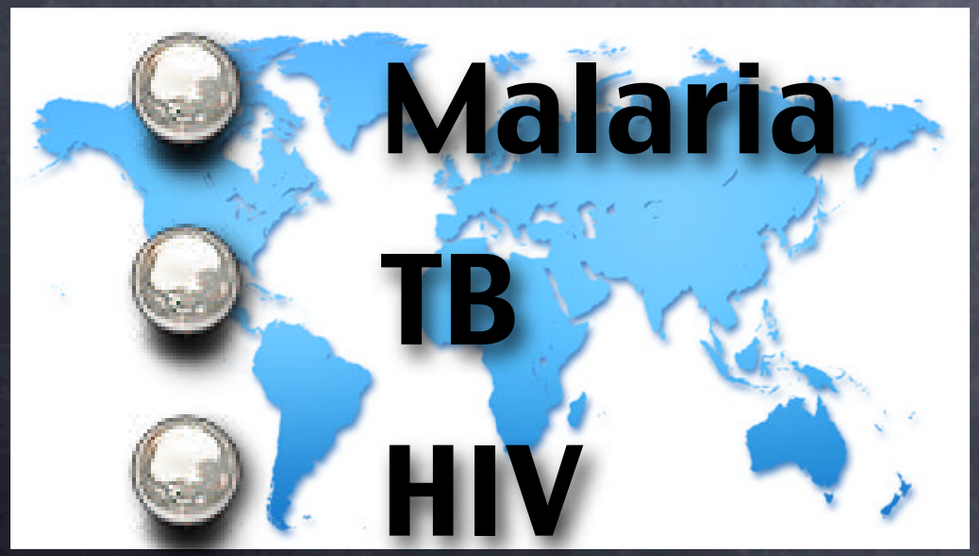




# Medical Tricorder


# Medical Tricorder

# The Future Begins




# The Future Begins

Jan 12, 2010
Haiti Earthquake 7.0M

Feb 26, 2010
Japan Earthquake 7.3M

Feb 27, 2010
Chile Earthquake 8.8M

Mar 4, 2010
Taiwan Earthquake 6.4M

Mar 15, 2010
Japan Earthquake 6.6M

# Stanford EE46

# Global Health Research Foundation

http://zwazosms.agilityhoster.com/iphone/

http://www.ghrf.org

## Medical Tricorder


# Medical Tricorder

http://www.slideshare.net/bess.ho

# Q&A