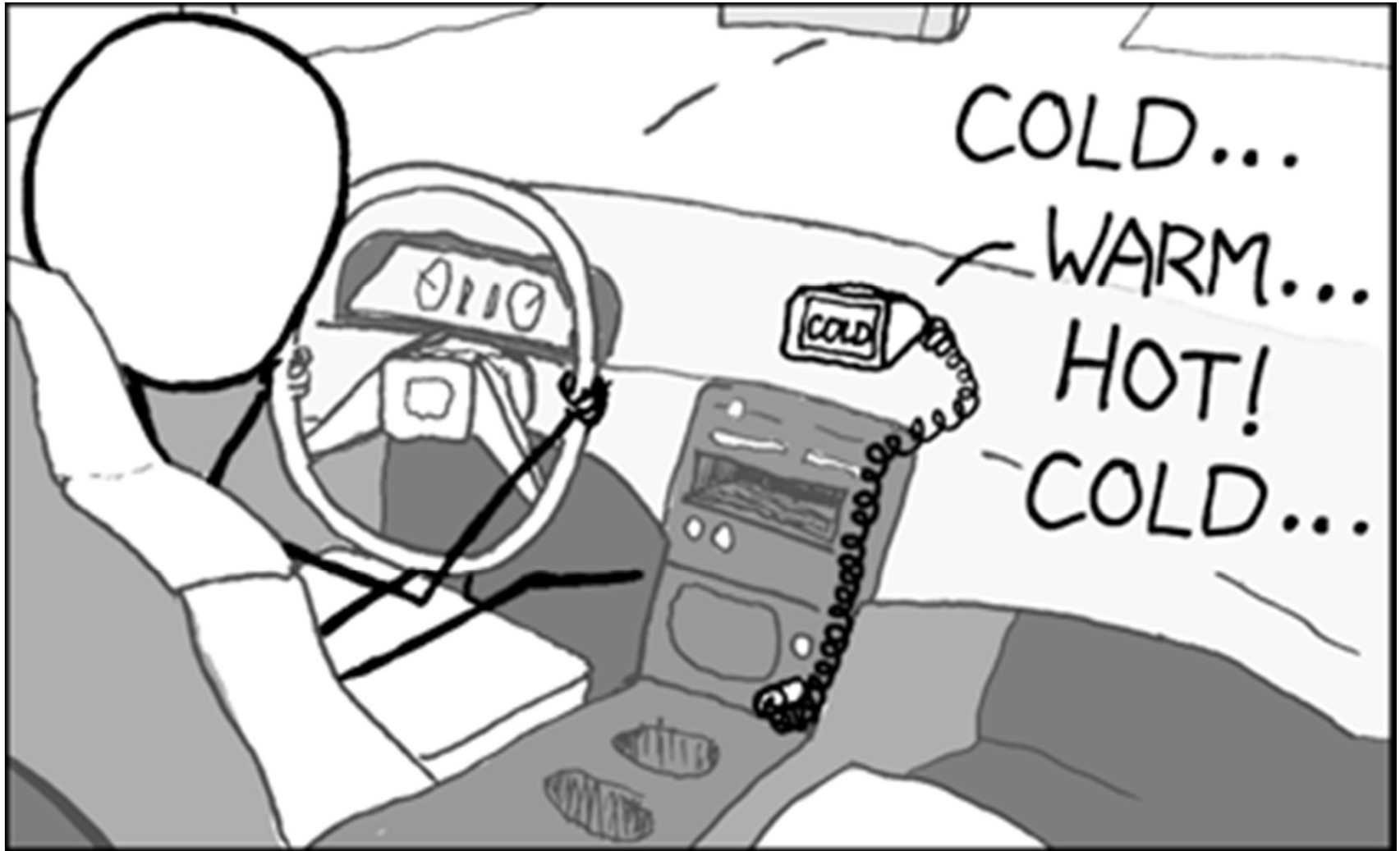


CS378 - Mobile Computing

Location

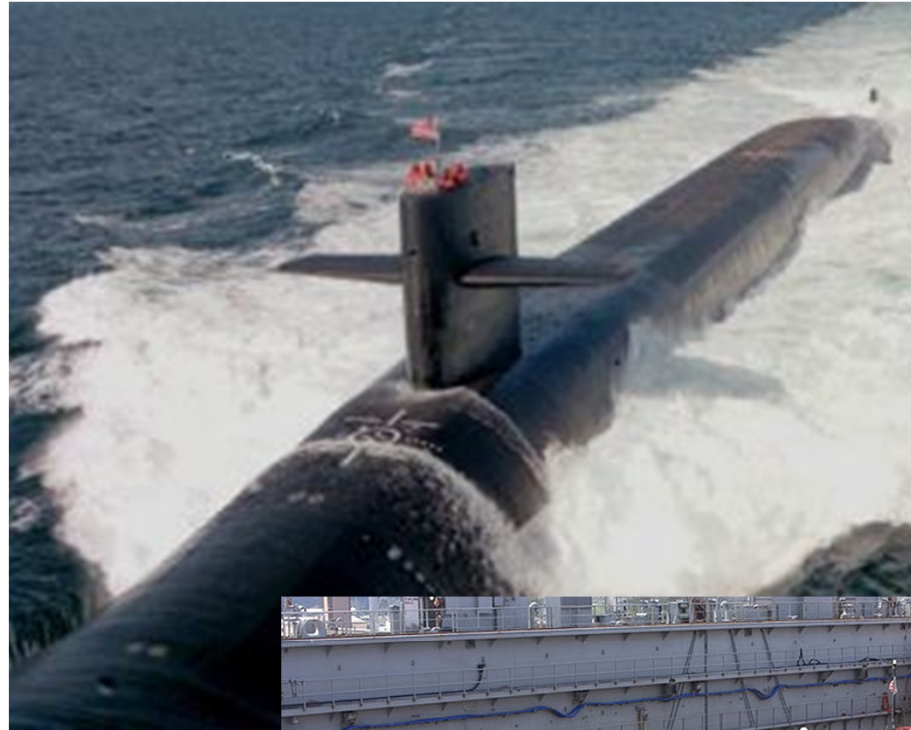
Cheap GPS



<http://xkcd.com/407/>

Location, Location, Location

- Dead reckoning
- radar fix
- visual fix
- Loran
- Omega
- Navsat
- GPS
- Active Sonar
- Inertial Navigation System



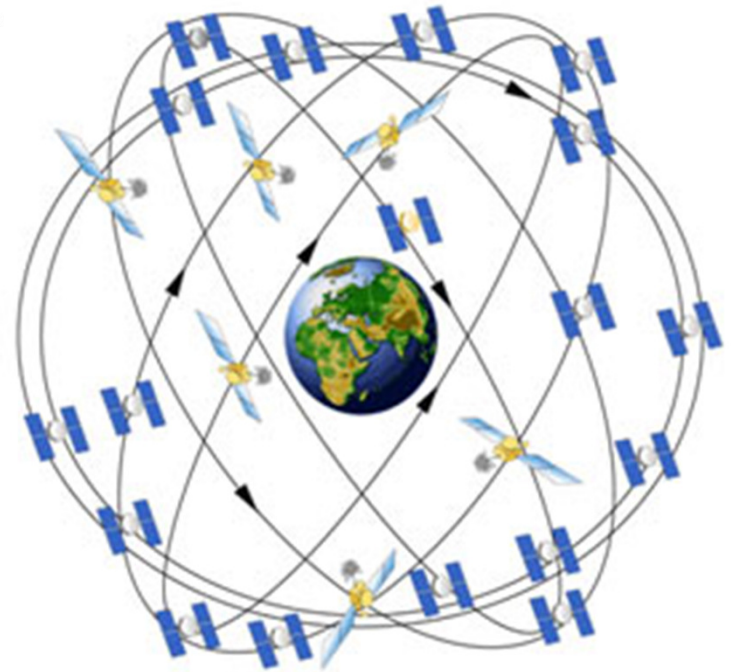
Global Positioning System

- GPS
- US System that provides position, navigation, and timing
- Space Segment, Control Segment, User Segment
- US Air Force develops, maintains, and operates the space segment and control segment



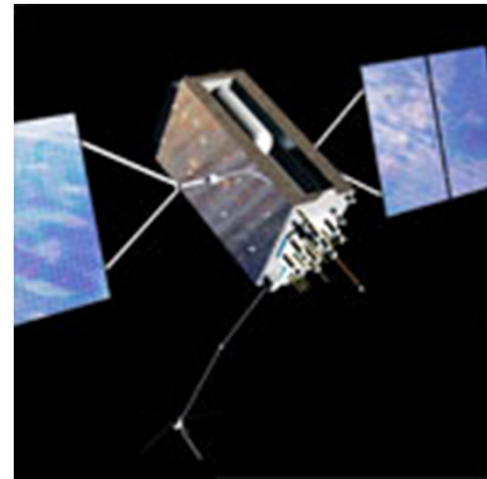
GPS Space Segment

- 24 core satellites
- medium earth orbit, 20k km above the earth
- 6 orbital planes with 4 satellites each
- generally 4 satellites in line of sight at any spot on the earth
- recently upgraded to 27 sats



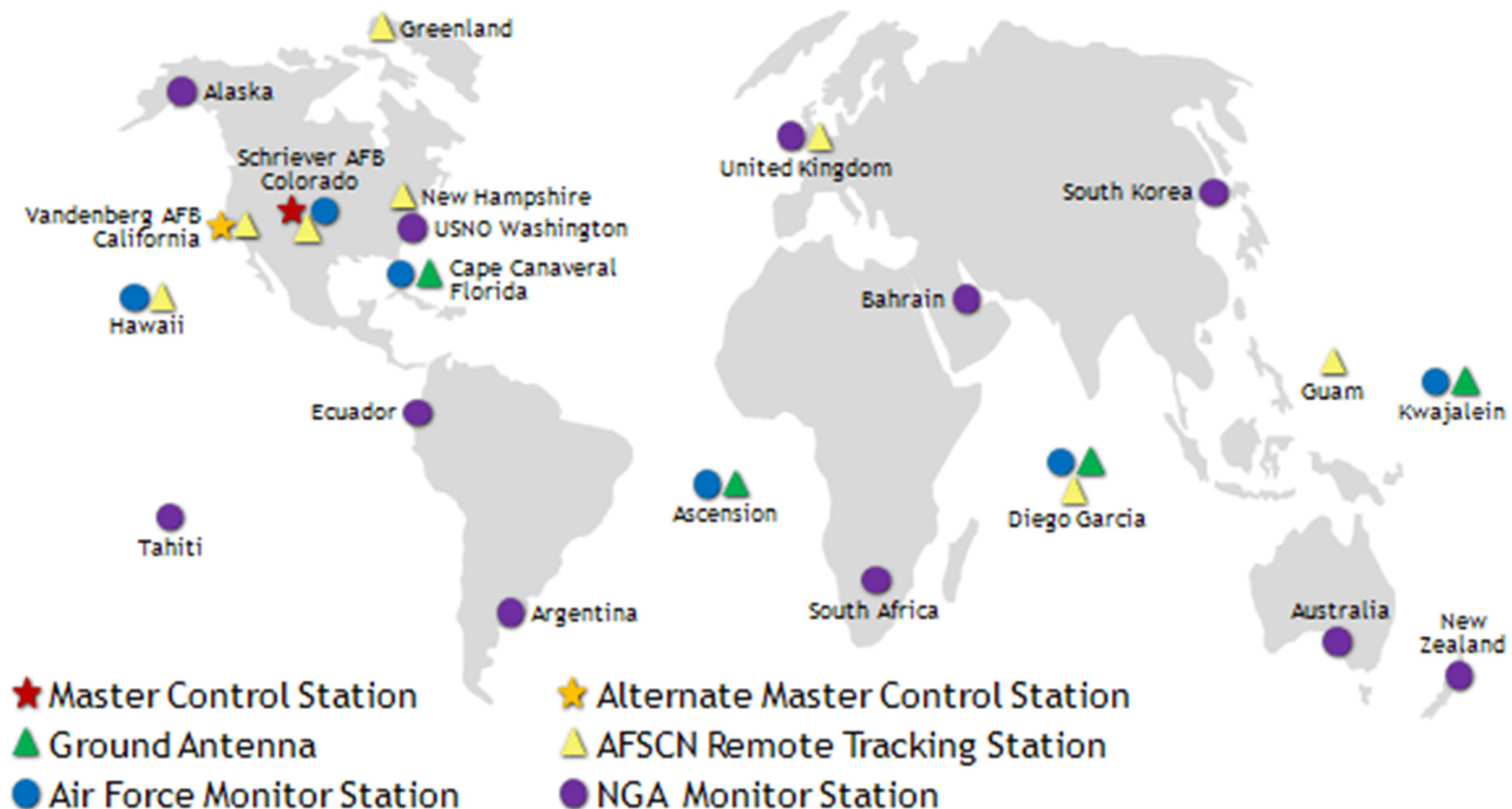
GPS Space Segment

- satellites circle the earth twice a day
- upgraded over time with different generations of satellites
- Current generation of satellites being developed by Lockheed - Martin



GPS Control Segment

- Ground facilities that
 - monitor transmissions, perform analysis, and send commands and data to satellites



GPS User Segment

- Onboard clocks with accuracy of 1 nanosecond (1 billionth of a second)
- Satellites transmit one way
- receiver calculates position and course by comparing time signals from multiple satellites with the known position of those satellites

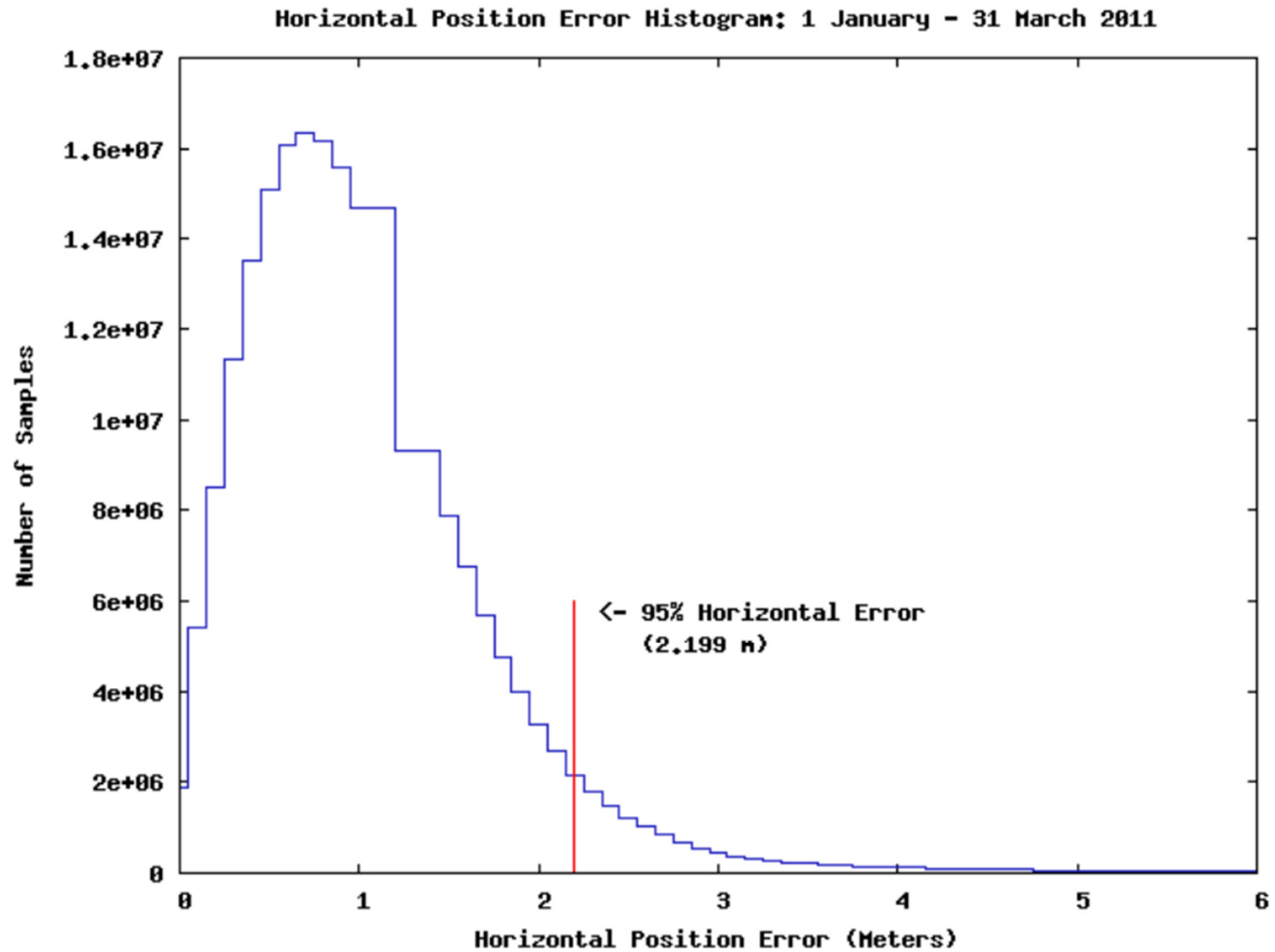
CRAZY PHENOMENON	IF IT WORKED, COMPANIES WOULD BE USING IT TO MAKE A KILLING IN...	ARE THEY?
REMOTE VIEWING	OIL PROSPECTING	
DOWSING		
AURAS	HEALTH CARE COST REDUCTION	
HOMEOPATHY		
REMOTE PRAYER		
ASTROLOGY	FINANCIAL/BUSINESS PLANNING	
TAROT		
CRYSTAL ENERGY	REGULAR ENERGY	
CURSES, HEXES	THE MILITARY	
RELATIVITY	GPS DEVICES	✓
QUANTUM ELECTRODYNAMICS	SEMICONDUCTOR CIRCUIT DESIGN	✓

EVENTUALLY, ARGUING THAT THESE THINGS WORK MEANS ARGUING THAT MODERN CAPITALISM ISN'T *THAT* RUTHLESSLY PROFIT-FOCUSED.

GPS User Segment

- accuracy easily within 5 - 10 meters
- precision requires accuracy of clocks and timing signal on the order of 20 nanoseconds
- the Special and General theories of Relativity must be taken into account to achieve the desired accuracy
- Special relativity predicts clocks on satellites go slower, on the order of 10 microseconds per day
- General relativity predicts the mass of the earth will also have an effect

GPS Accuracy

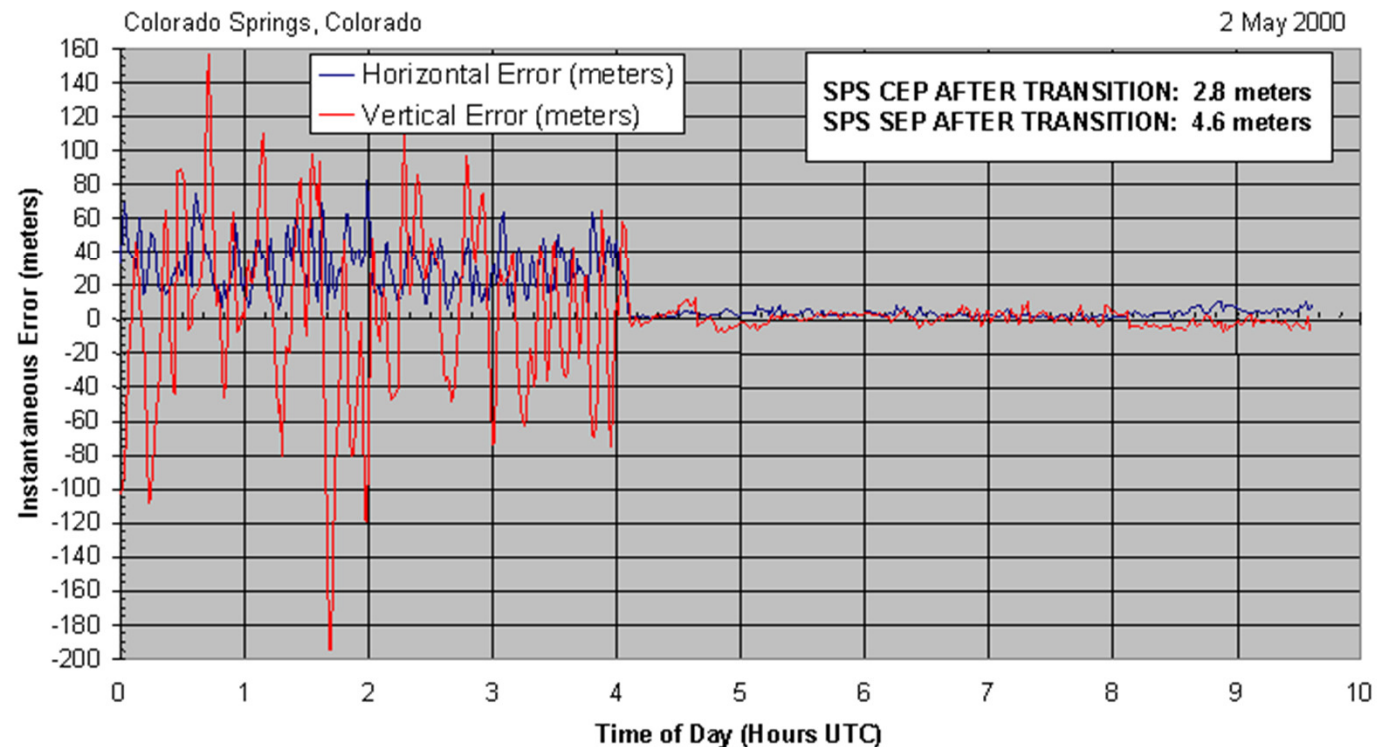


GPS Accuracy

- Selective Availability: intentional degradation of signals for civilian use
 - ended in 2000



SA Transition -- 2 May 2000



GPS Accuracy

- civilian GPS: aka SPS
- military GPS: aka PPS
- military broadcasts on two frequencies, civilian only one
- "This means military users can perform *ionospheric correction*, a technique that reduces radio degradation caused by the Earth's atmosphere. With less degradation, PPS provides better accuracy than the basic SPS. "

Android and Location

- Obtaining User Location
- GPS
 - most accurate but,
 - only works OUT doors
 - quickly consumes battery power
 - delay in acquiring satellites or requiring if lost



Android and Location

- Other sources of location for Android device include:
- cell-ID
- Wi-Fi networks
- Android can use GPS and the Network Location Provider which combines cell-ID and Wi-Fi data

Finding Location

- Add appropriate permission to AndroidManifest.xml
- Get instance of LocationManager using getSystemService method
- Choose location provider
- Implement a LocationListener class
- Call requestLocationUpdates method with chosen provider so LocationListener start receiving location information

AndroidManifest.xml

- User Permission in manifest

```
<manifest ... >
  <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
  ...
</manifest>
```

- Options: ACCESS_FINE_LOCATION or ACCESS_COARSE_LOCATION
- ACCESS_COARSE_LOCATION for use of NETWORK_PROVIDER using cell-ID and Wi-Fi
- ACCESS_FINE_LOCATION: GPS or NETWORK_PROVIDER

Location Manager

- Obtain Location Manager
- Implement class that implements LocationListener interface

Public Methods	
abstract void	<code>onLocationChanged (Location location)</code> Called when the location has changed.
abstract void	<code>onProviderDisabled (String provider)</code> Called when the provider is disabled by the user.
abstract void	<code>onProviderEnabled (String provider)</code> Called when the provider is enabled by the user.
abstract void	<code>onStatusChanged (String provider, int status, Bundle extras)</code> Called when the provider status changes.

Obtaining Locations

- Register the `LocationListener` to receive location updates
- `locationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, 0, 0, locationListener);`
 - provider: name of provider to register with
 - minTime: the minimum time interval for notifications, in milliseconds. only a hint to conserve power, and actual time between location updates may be greater or lesser than this value.
 - minDistance: min distance interval for notifications in meters
 - the listener itself

requestLocationUpdates

- More on arguments
- 0 for minTime AND minDistance indicate obtain updates as frequently as possible
- for *background services* recommended minTime \geq 60,000 ms to avoid consuming too much power with the GPS or Wi-Fi receivers
- 60,000 ms = 60 seconds!
- clearly less non background apps

Location Listener

```
// Acquire a reference to the system Location Manager
LocationManager locationManager = (LocationManager) this.getSystemService(Context.LOCATION_SERVICE)

// Define a listener that responds to location updates
LocationListener locationManager = new LocationListener() {
    public void onLocationChanged(Location location) {
        // Called when a new location is found by the network location provider.
        makeUseOfNewLocation(location);
    }

    public void onStatusChanged(String provider, int status, Bundle extras) {}

    public void onProviderEnabled(String provider) {}

    public void onProviderDisabled(String provider) {}
};

// Register the listener with the Location Manager to receive location updates
locationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, 0, 0, locationManager);
```


Simple Location Program

- Just to demonstrate capabilities
- After setting up listener show all providers
- mgr is LocationManager

```
/** Write information from all location providers */  
private void dumpProviders() {  
    List<String> providers = mgr.getAllProviders();  
    for (String provider : providers) {  
        dumpProvider(provider);  
    }  
}
```

Properties of Location Providers

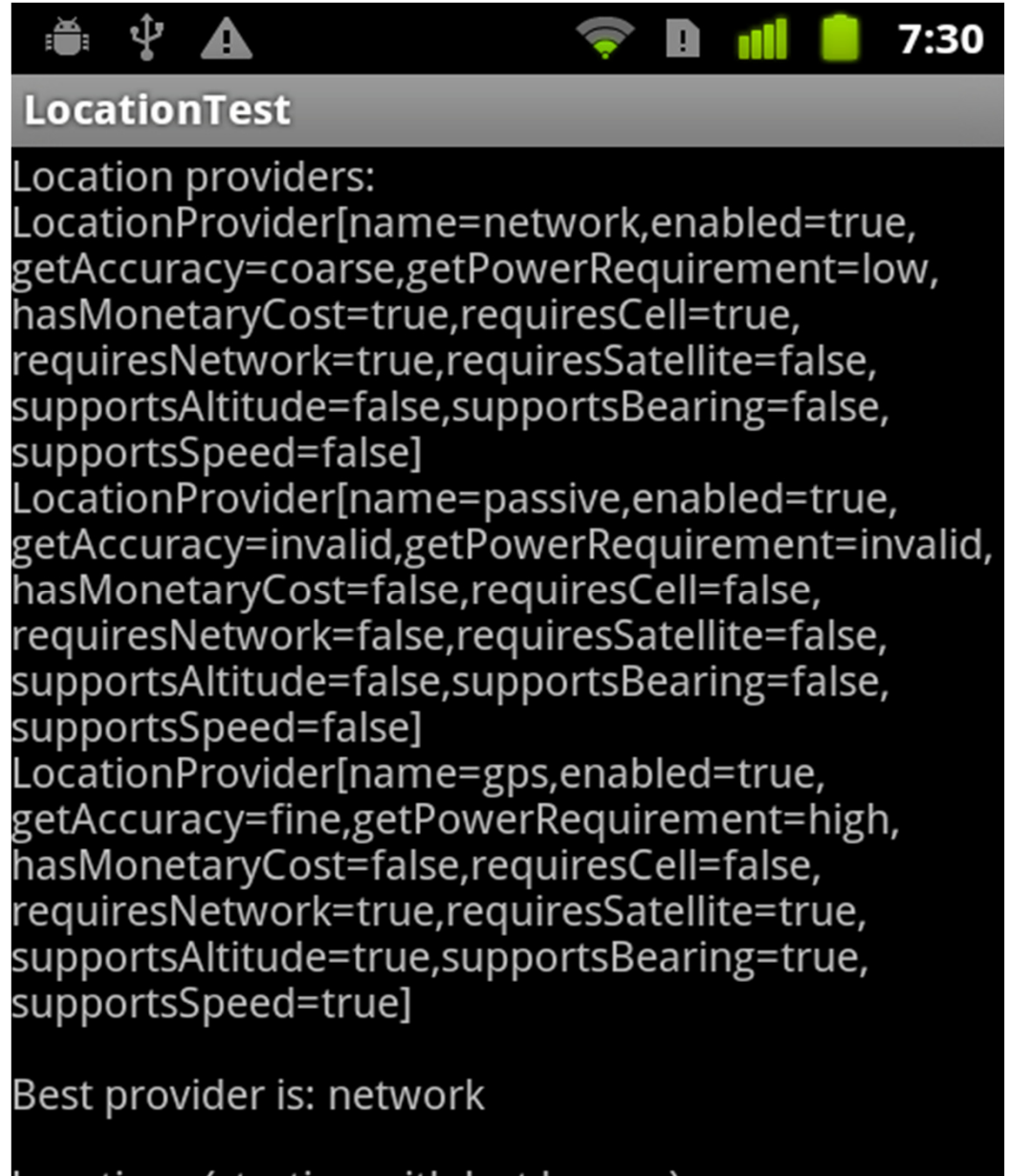
- name
- enabled
- accuracy
- power requirements
- monetary cost
- requires cell
- requires network
- requires satellite
- supports altitude
- supports bearing
- supports speed

Bearing

- direction
- 360 degrees
- degrees east of north
- 0 = north
- 90 = east
- 180 = south
- 270 = west

Program Output

- network
- gps
- passive
 - use location updates requested by other applications or services

A screenshot of an Android application interface. At the top is a status bar with icons for a robot, USB, a warning triangle, Wi-Fi, signal strength, battery, and the time 7:30. Below the status bar is a title bar with the text 'LocationTest'. The main content area has a black background with white text. It lists 'Location providers:' followed by three entries for 'network', 'passive', and 'gps'. Each entry shows various attributes like 'enabled', 'getAccuracy', 'getPowerRequirement', 'hasMonetaryCost', 'requiresCell', 'requiresNetwork', 'requiresSatellite', 'supportsAltitude', 'supportsBearing', and 'supportsSpeed'. At the bottom, it states 'Best provider is: network'.

```
LocationTest

Location providers:
LocationProvider[name=network,enabled=true,
getAccuracy=coarse,getPowerRequirement=low,
hasMonetaryCost=true,requiresCell=true,
requiresNetwork=true,requiresSatellite=false,
supportsAltitude=false,supportsBearing=false,
supportsSpeed=false]
LocationProvider[name=passive,enabled=true,
getAccuracy=invalid,getPowerRequirement=invalid,
hasMonetaryCost=false,requiresCell=false,
requiresNetwork=false,requiresSatellite=false,
supportsAltitude=false,supportsBearing=false,
supportsSpeed=false]
LocationProvider[name=gps,enabled=true,
getAccuracy=fine,getPowerRequirement=high,
hasMonetaryCost=false,requiresCell=false,
requiresNetwork=true,requiresSatellite=true,
supportsAltitude=true,supportsBearing=true,
supportsSpeed=true]

Best provider is: network
```

Location Data

- onLocationChange method in the LocationListener receives Location objects
- toString shown
- latitude, longitude, timestamp, possibly altitude, speed, and bearing

```
LocationTest

Locations (starting with last known):

Location[mProvider=network,
mTime=1333326955116,mLatitude=30.5158771,
mLongitude=-97.7139099,mHasAltitude=false,
mAltitude=0.0,mHasSpeed=false,
mSpeed=0.0,mHasBearing=false,
mBearing=0.0,mHasAccuracy=true,
mAccuracy=47.0,mExtras=Bundle[mParcelledData.
dataSize=148]]

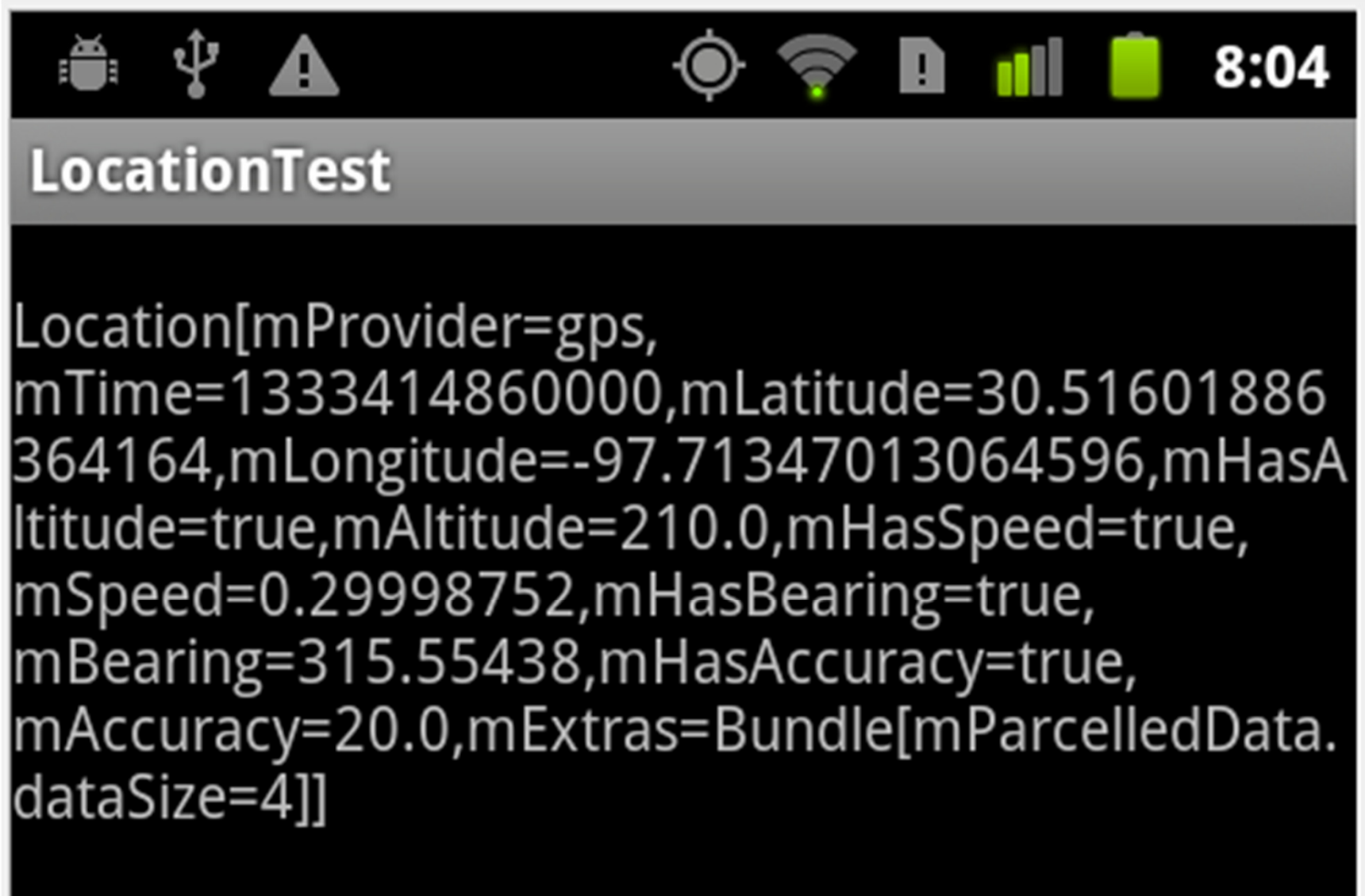
Location[mProvider=network,
mTime=1333327587135,mLatitude=30.5159603,
mLongitude=-97.7133946,mHasAltitude=false,
mAltitude=0.0,mHasSpeed=false,
mSpeed=0.0,mHasBearing=false,
mBearing=0.0,mHasAccuracy=true,
mAccuracy=35.0,mExtras=Bundle[mParcelledData.
dataSize=148]]

Location[mProvider=network
```

Sample GPS Locations



Sample GPS Locations



How does Wi-Fi Fix Location?

- Use to use StreetView cars
- Now, use the devices themselves to map locations to wi-fi spots
- Apple and Microsoft do the same thing
- default on dev phones was checked



Google Location Services

