Aalto University
School of Science

Energy-efficient Mobile Sensing

T-110.6120 Spring 2014

Yu Xiao

28.4.2014

Agenda

- **Sensors on smartphones**
- **Energy-efficient Positioning**
- **Energy-efficient Trajectory Tracking**
- **Energy-efficient User State Recognition**
- **Sensor Hub**

Which sensors are available on a smartphone?

Example: getting sensor information using Android 4.2.2 SDK

```
SensorManager mSensorManager =
(SensorManager)getSystemService(SENSOR_SERVICE);
List<Sensor> mSensorList = mSensorManager.getSensorList(Sensor.TYPE_ALL);
String sResult = "";


for (Sensor mSensor: mSensorList){
    sResult += String.format("Name:%s, maxRange:%f, Resolution:%f, Power:%f;\r\n",
mSensor.getName(), mSensor.getMaximumRange(), mSensor.getResolution(),
mSensor.getPower());
}
```

For each sensor, list sensor name, maximum range, resolution, and power requirements

Example Output

Name:LIS303DLHC 3-axis Accelerometer, maxRange:39.226601, Resolution:0.009577, Power:0.230000;

Name:AK8963 3-axis Magnetic field sensor, maxRange:4915.200195, Resolution:0.060000, Power:0.280000;

Name:iNemo Orientation sensor, maxRange:360.000000, Resolution:0.100000, Power:13.000000;

Name:Light sensor, maxRange:10000.000000, Resolution:1.000000, Power:0.750000;

Name:Proximity sensor, maxRange:5.000000, Resolution:5.000000, Power:0.750000;

Name:L3G4200D Gyroscope sensor, maxRange:34.906586, Resolution:0.001222, Power:6.100000;

Name:iNemo Gravity sensor, maxRange:9.806650, Resolution:0.153281, Power:0.200000;

Name:iNemo Linear Acceleration sensor, maxRange:39.226601, Resolution:0.009577, Power:0.200000;

Name:iNemo Rotation Vector sensor, maxRange:1.000000, Resolution:0.000000, Power:6.100000;

Sensors Managed by SensorManager

- **Motion Sensors:** measure acceleration forces and rotational forces along three axes (x, y, and z)
  - ✓ Accelerometers: acceleration force in m/s^2
  - ✓ Gravity sensors: force of gravity in m/s^2
  - ✓ Gyroscopes: a device's rate of rotation in rad/s
  - ✓ Rotaional vector sensors: the orientation of a device

Sensors Managed by SensorManager

- **Environmental Sensors:** measure various environmental parameters, such as ambient air temperature and pressure, illumination, and humidity
  - ✓ Barometers
  - ✓ Photometers
  - ✓ Thermometers

- **Position Sensors:** measure the physical position of a device**.**
  - Orientation sensors
  - Magnetometers

Any other sensors available on Smartphones?

Sensors in Smartphones

Samsung Galaxy S4

- Dual Cameras: photo/video (1080p@30fps)
- Microphone
- Position: GPS, Wi-Fi, cellular, Bluetooth, NFC
- Accelerometer, Gyroscope, Proximity,Compass
- Barometer
- Temperature
- Humidity
- Gesture

Apple iPhone 5

- Dual Cameras: photo/video (1080p@30fps), panorama
- Microphone
- Position: GPS, Wi-Fi, Cellular, Bluetooth
- Accelerometer
- Gyroscope
- Proximity
- Compass
- Ambient light sensor

Sensors in Wearable Devices

Google Glasses

- 5-megapixel camera, 720p video recording
- Microphone
- GPS
- Wi-Fi 802.11b/g
- Bluetooth
- Gyroscope
- Accelerometer
- Compass
- Ambient light sensing and proximity sensor

Samsung Galaxy Gear

- 1.9-megapixel camera, 720p video recording
- Microphone
- Bluetooth
- Gyroscope
- Accelerometer
- Pedometer

More Sensors on Smartphones

- **Envionmental Sensors**
  - ✓ Cameras
  - ✓ Microphone
- **Position Sensors**
  - ✓ GPS
  - ✓ Cellular
  - ✓ Wi-Fi
  - ✓ Bluetooth
  - ✓ NFC

Not just for wireless data transmission (Cellular, Wi-Fi, Bluetooth, NFC)

Two Ways of Data Collection

- **Periodic sampling**
  - ✓ Get 3-axes accelerometer readings at 1Hz
  - ✓ Scan Wi-Fi APs every 5 minutes
  - ✓ Take a photo every 1 minute

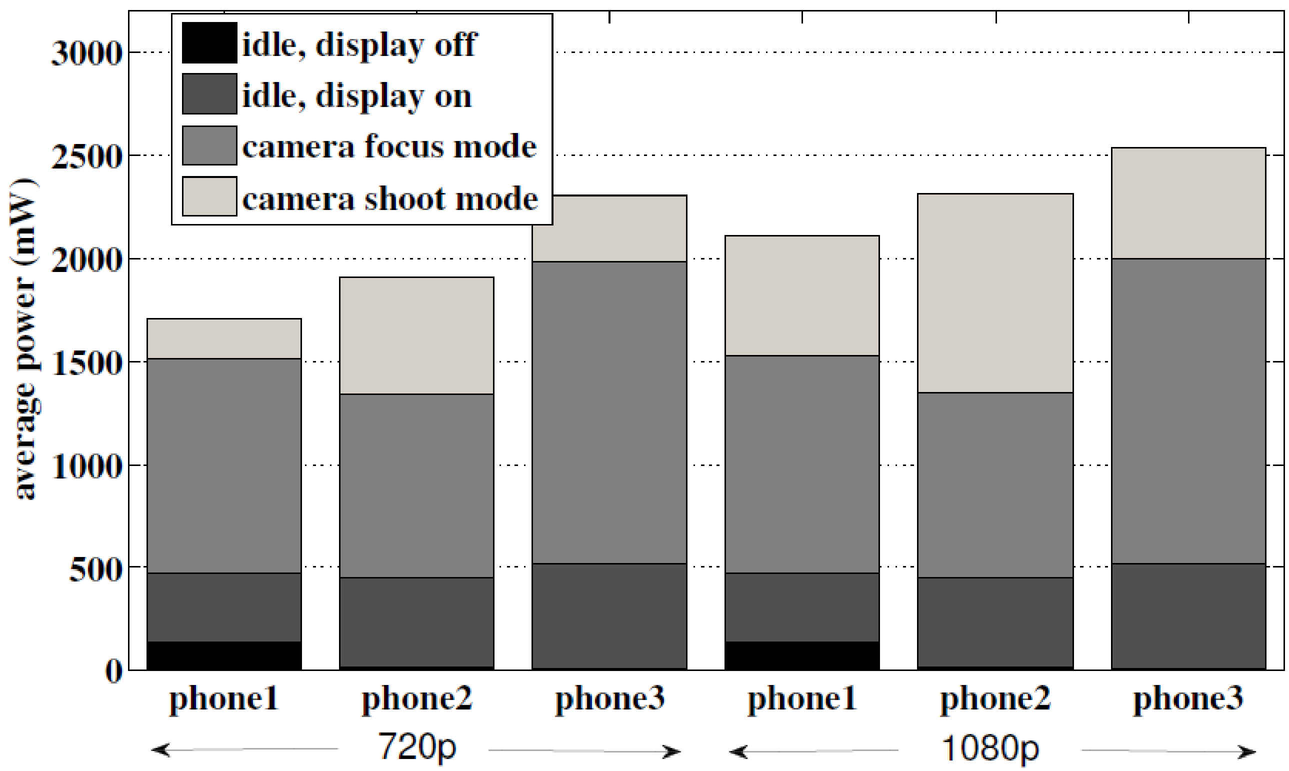
- **Listen for events that indicate changes in status**
  - ✓ GPS: minimal distance between location updates
  - ✓ Get Wi-Fi SSID when Wi-Fi interface is connected

Is it expensive in terms of power consumption to collect sensor data?

Sample Energy and Power Measures for Sensors of Samsung Galaxy S2

Sensor	Switch ON	Switch OFF	Sampling	Idle	Pre-sampling
Accelerometer	-	-	21 mW	-	-
Gravity	-	-	25 mW	-	-
L.Acceleration	-	-	25 mW	-	-
Magnetometer	-	-	48 mW	20 mW	-
Orientation	-	-	49 mW	20 mW	-
Rotation	-	-	50 mW	21 mW	-
Gyroscope	-	-	130 mW	22 mW	44 mJ
Microphone	123 mJ	36 mJ	101 mW	-	-
GPS	77 mJ	-	176 mW	-	198 mW

Power Consumption of Camera



Phone 1&2: IPS LCD
Phone 3: AMOLED

Energy Management of Sensors

- **Shutting down unnecessary sensors**
- **Selecting sensors with a low power consumption whenever possible**
- **Optimizing Sensor Duty Cycling (i.e., sensors will adopt periodic sensing and sleeping instead of being sampled continuously)**

Potential Tradeoffs: Granularity, accuracy vs. Power consumption

Android Location Services

Location Services sends the current location to your app through a location client.

Android has two location permissions:

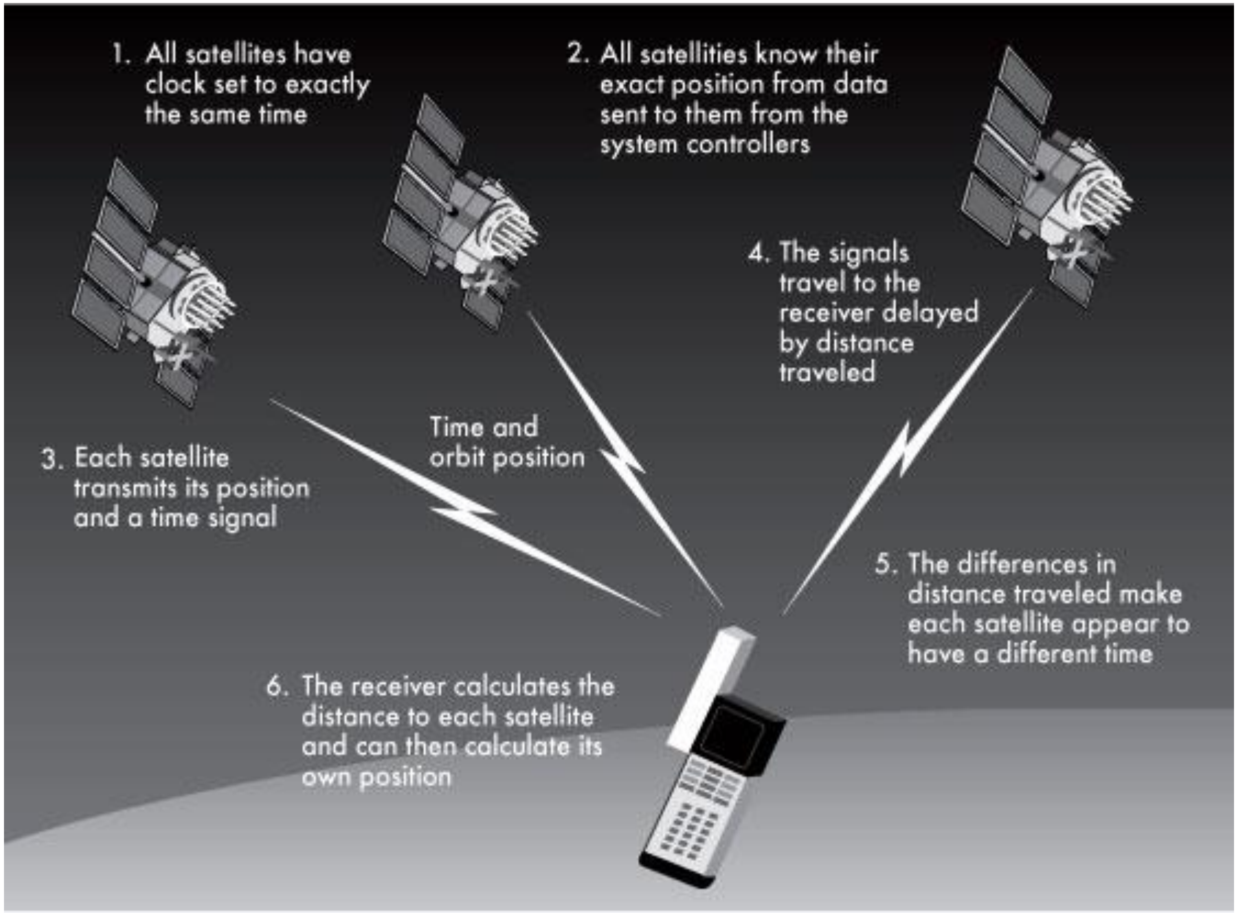
*1) android.permission.**ACCESS_FINE_LOCATION***

Allows an app to access precise location from location sources such as **GPS, cell towers, and Wi-Fi**.

*2) android.permission.**ACCESS_COARSE_LOCATION***

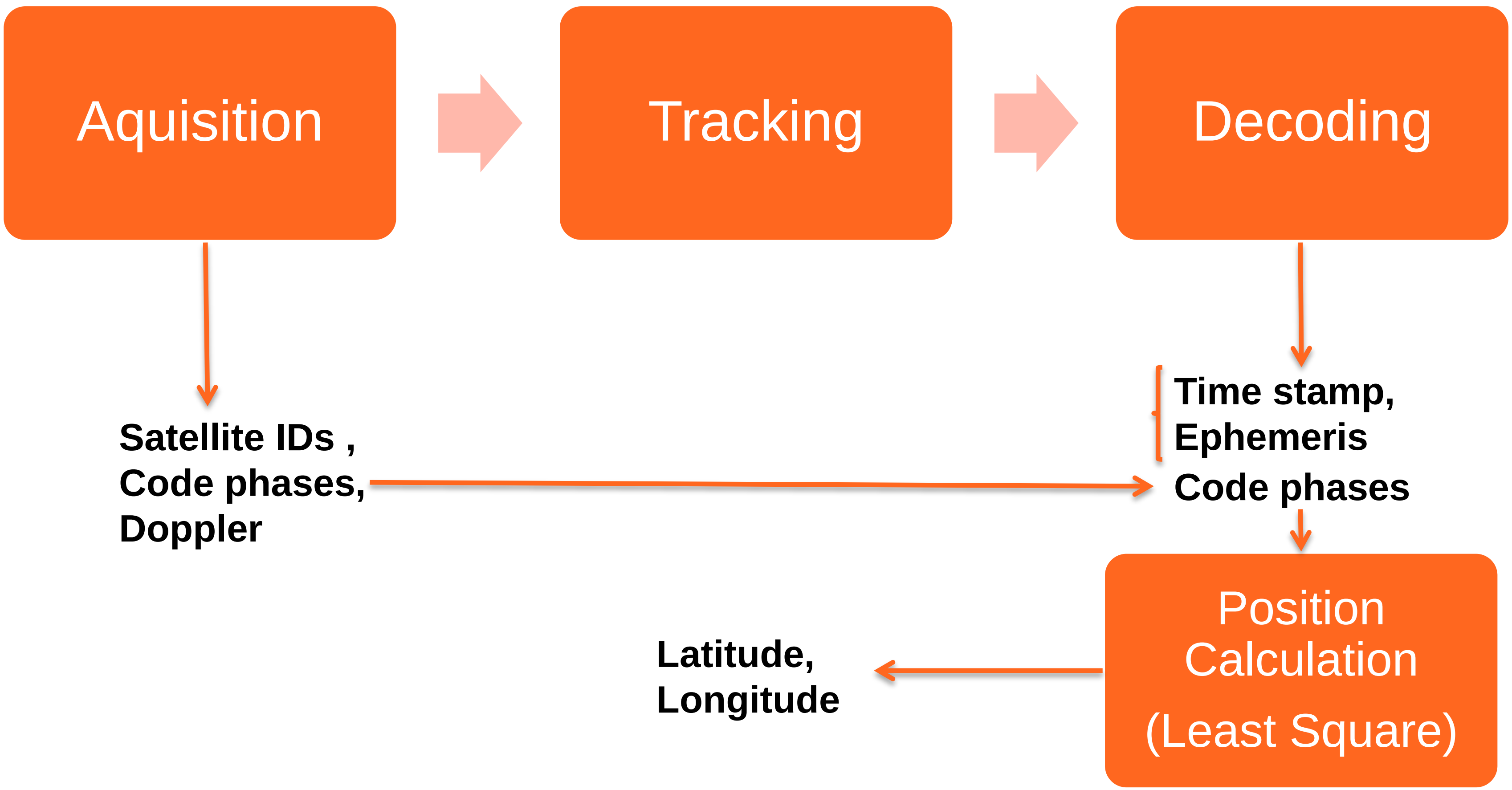
Allows an app to access approximate location derived from network location sources such as **cell towers and Wi-Fi**.

GPS



Source: http://mason.gmu.edu/~ttruong6/how.html

GPS Receiving



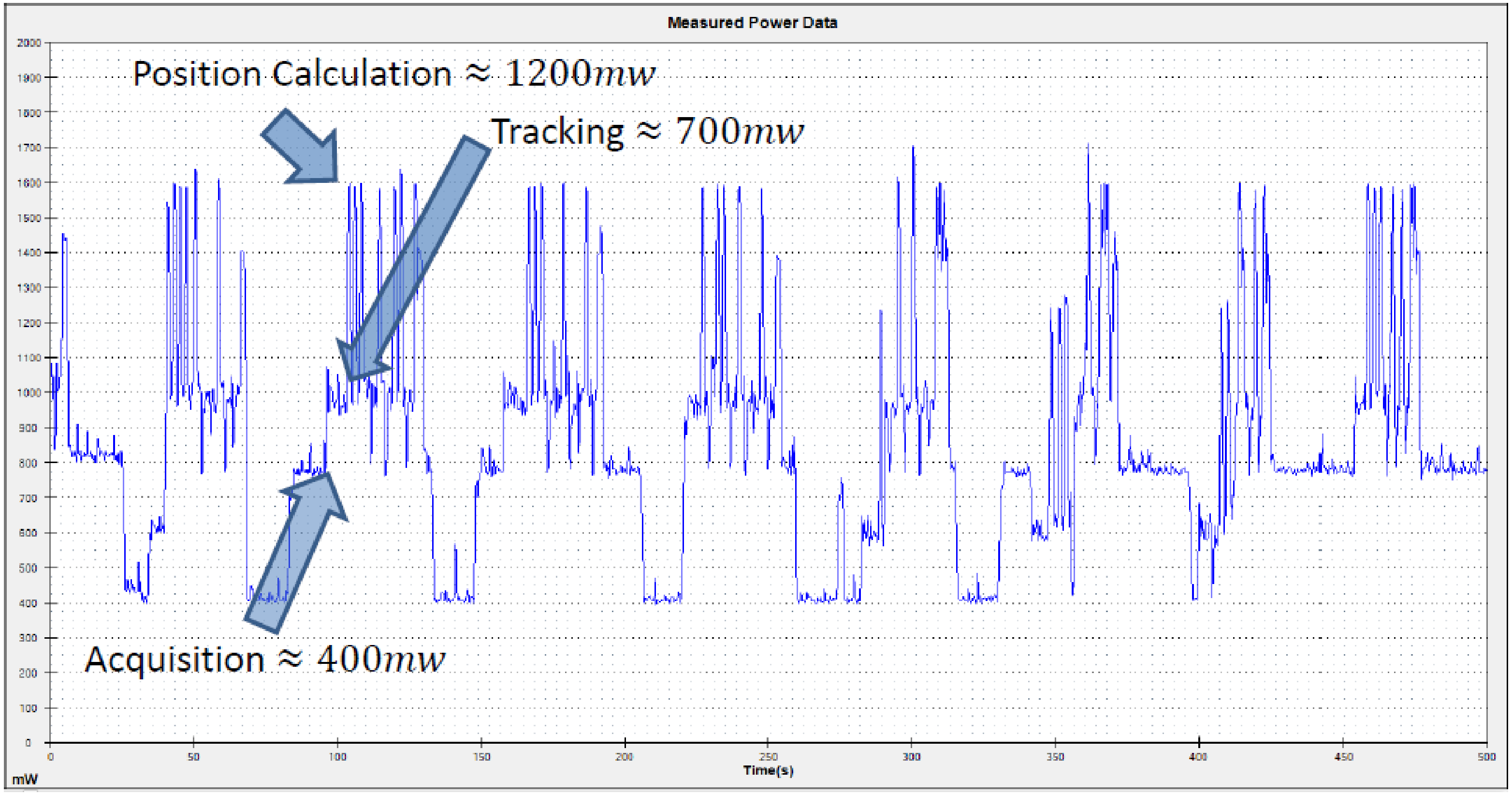
- **Acquisition**
  - ✓ Each satellite encodes its signal (CDMA encoded) using a satellite-specific coarse/acquisition(C/A) code
  - ✓ When a GPS receiver first starts up, it needs to detect what satellites are in view (comparing C/A codes in the received signal with each known C/A code templates)
  - ✓ Compute-intensive: it must search through 30+ frequency bins times 8,000+ code phase possibilities for each single satellite

- **Tracking**
  - ✓ Adjust previously acquired Doppler frequency shifts and code phases to the new ones
  - ✓ Once a GPS produces its first location fix, subsequent locatino estimates become fast
  - ✓ Relatively inexpensive process
  - ✓ After 30 seconds of non-tracking, the GPS receiver has to start all over again

- **Decoding: based on the signals and data packets sent from the satellites, the GPS receiver can infer**
  - ✓ A precise time T
  - ✓ A set of visible GNSS satellites and their locations at time T
  - ✓ The distance from the receiver to each satellite at time T

- **Position calculation**
  - ✓ Using constraint optimization techniques such as Least Squares minimization
  - ✓ Requires powerful CPU

GPS Power Consumption

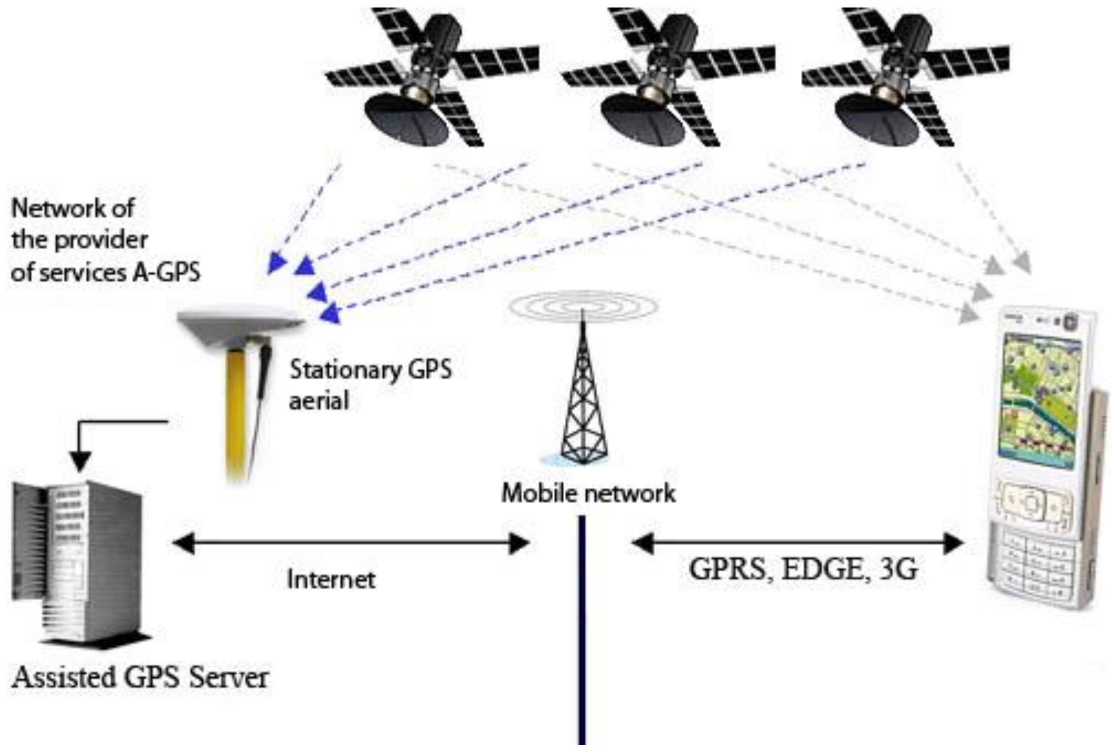


Source:http://research.microsoft.com/en-us/people/liuj/co-gps-sensys12.pdf

How to Speed Up Aquisition

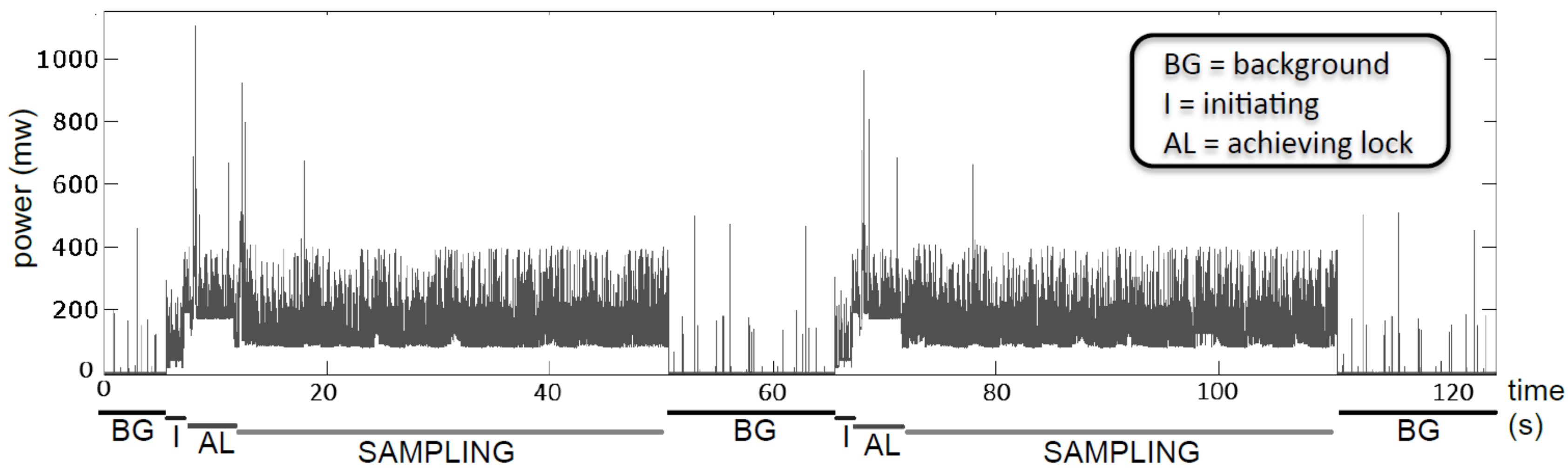
- **Cold Start:** the receiver has no prior knowledge of the satellites and its own location, it has to search the entire space
- **Warm Start:** the receiver has a previous lock to the satellites, it can start from the previous Doppler shift and code phases and search around them
- **Hot Start:** the previous satellite locks are within a second, the receiver can skip the acquisition process and start directly from tracking to refine the Doppler and code phases
- **A-GPS:** the infrastructure provides the up-to-date satellites' trajectory so that the GPS receiver does not have to decode them from satelite signals

A-GPS



Source: http://gps-response.com/61-a-gps.html

Power Measurement from GPS sensor



Aquisition through A-GPS takes around 6 seconds

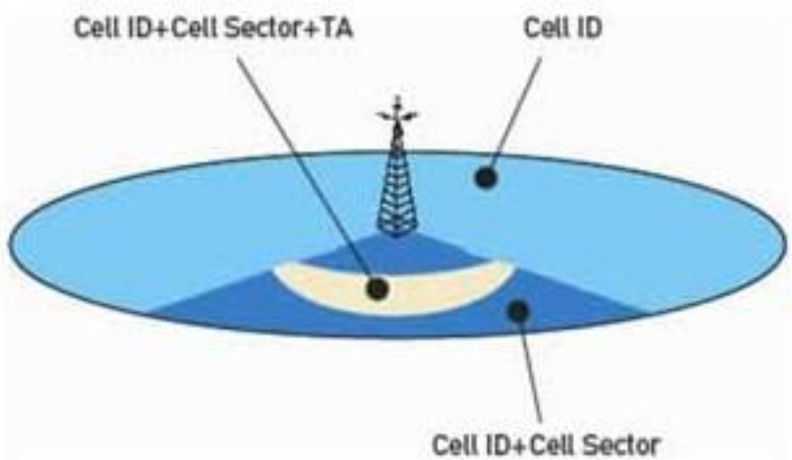
GPS vs. A-GPS

	GPS	A-GPS
Stands for	Global Positioning System	Assisted Global Positioning System
Source of triangulation information	Radio signals from GPS satellites	Radio signals from satellites and assistance servers e.g. mobile network cell sites
Reliability	GPS devices can determine location coordinates to within 1 meter accuracy	Location determined via A-GPS are slightly less accurate than GPS
Speed	GPS devices may take several minutes to determine their location because it takes longer to establish connectivity with 4 satellites.	A-GPS devices determine location coordinates faster because they have better connectivity with cell sites than directly with satellites.
Cost	GPS devices communicate directly with satellites for free. There is no cost of operation once the device is paid for.	It costs money to use A-GPS devices on an ongoing basis because they use mobile network resources.

Android App – GPS Status & Toolbox

https://play.google.com/store/apps/details?id=com.eclipsim.gpsstatus2&hl=en

Cell ID based Location Tracking



TA(Timing Advance): the length of time a signal takes to reach the base station from a phone

Cell ID based localization

- **For example, from Android phones, you can get**
  - ✓ GSM Cell ID (CID)
  - ✓ GSM location area code(LAC)
  - ✓ Network Type
  - ✓ Mobile Country Code (MCC)
  - ✓ Mobile Network Code(MNC)

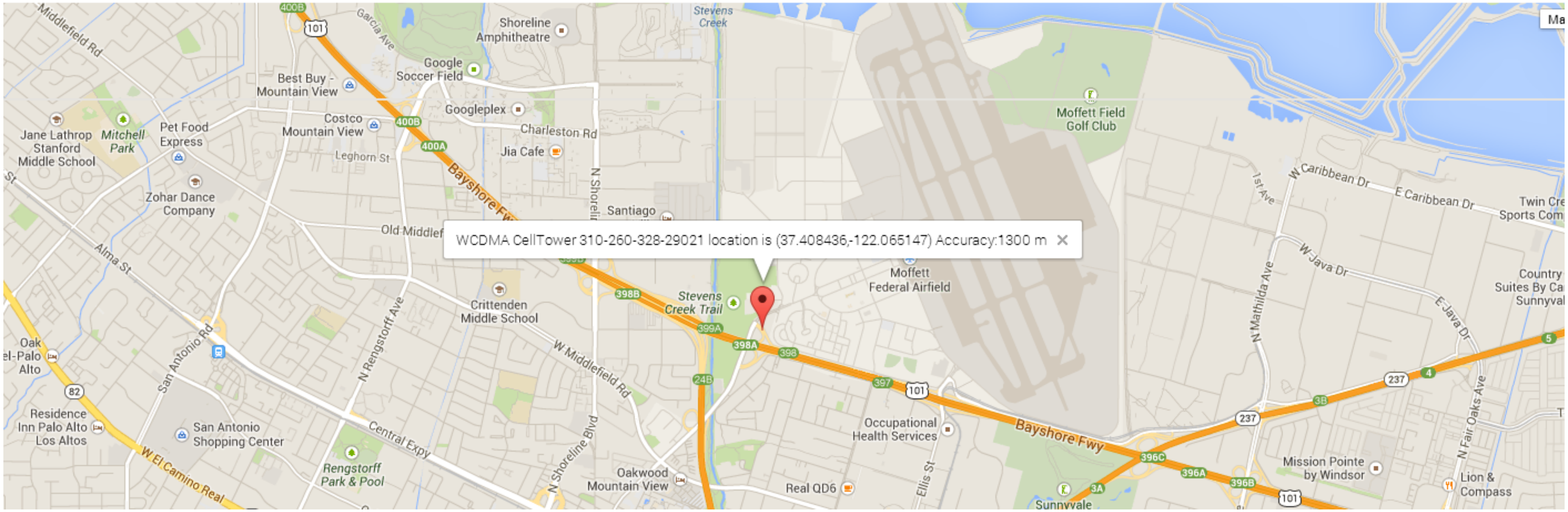
www.cell2gps.com

CellTower Locator

Track down a GSM/WCDMA cell phone online using LAC (Location Area Code) and Cell ID ,track down a CDMA/CDMA2000 cell phone online using SID,NID and BID, and display its location on Google Maps.
* indicates required.

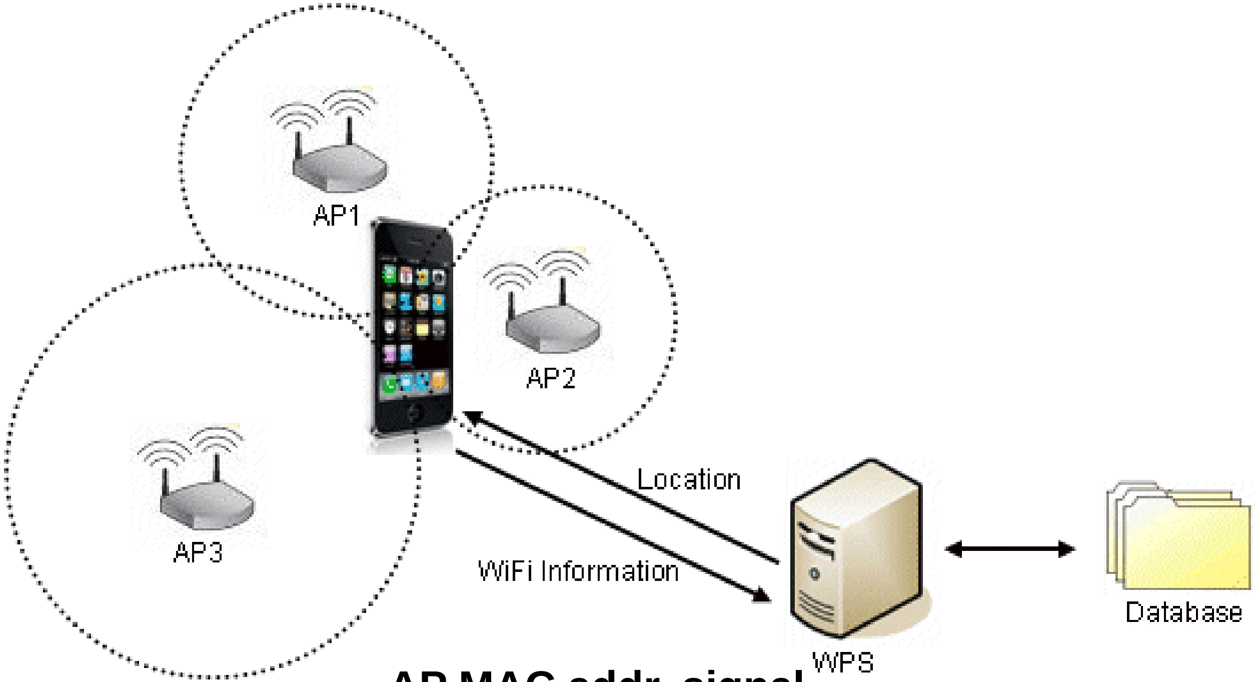
MCC	310	mcc list
MNC	260	mnc list
Network	GSM	
LAC*	328	
CellID*	29021	

Locate it



Wi-Fi Positioning

- **Fingerprinting**



AP MAC addr, signal strength, SSID, channel, etc

Workflow of Wi-Fi Fingerprinting

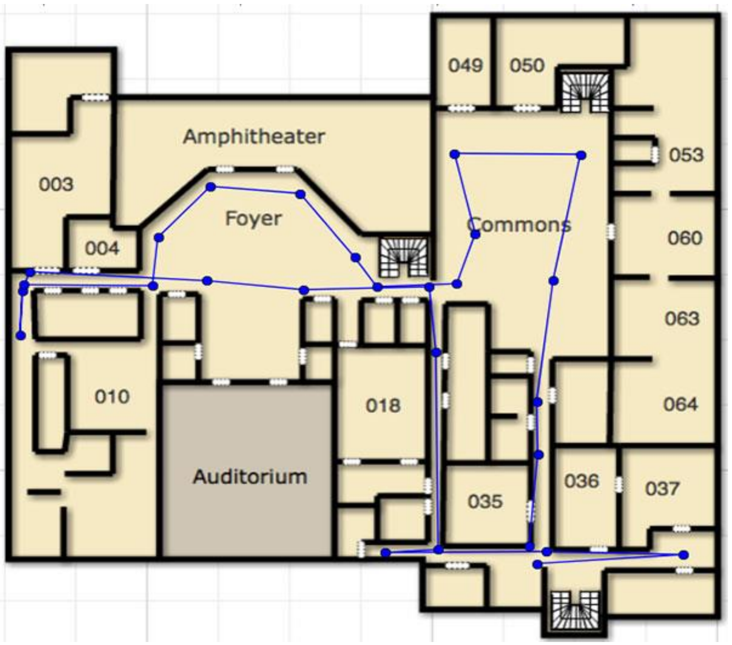
- Divide an area into presence areas and collect the signal strength from each area related to all the access points. The collected data is saved in a database.
- When trying to locate a phone, scan the Wi-Fi APs available and collect the information of APs
- Compare the information with the pre-collected information stored in the database, and determine the most possible position

Scanning APs requires significant amount of energy

More Advanced Indoor Positioning

Example: WifiSlam (acquired by Apple in 2013)

- **Wi-Fi + Accelerometer**
- Using pattern recognition and machine learning to draw **correlations between data gathered by *all* of the sensors in a device**.



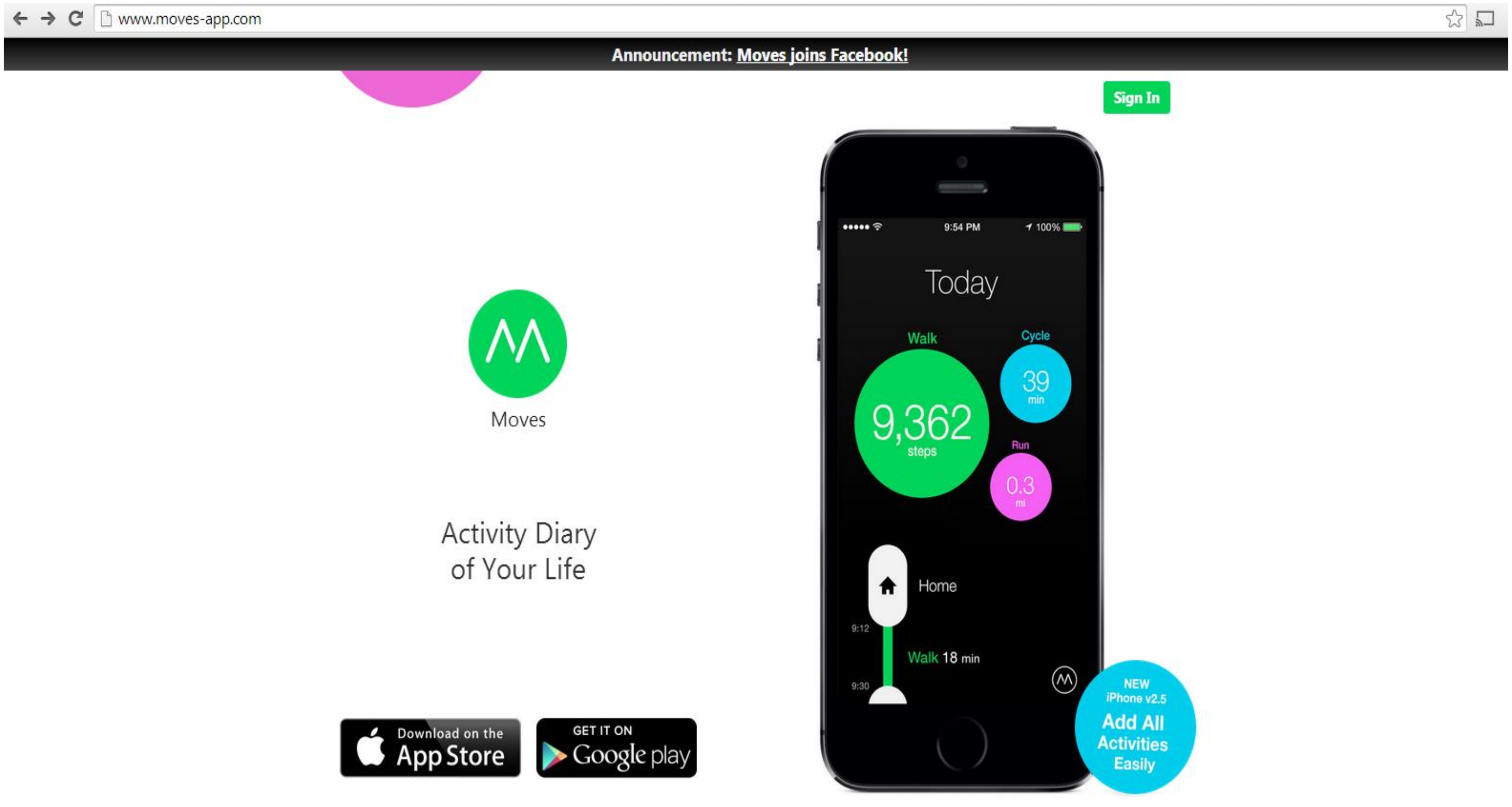
Summary of Position Sensors

- **GPS**
- **A-GPS**
- **Cell ID**
- **Wi-Fi**

Which ones to use depends on where you are(indoor/outdoor, urban/rural), availability of network infrastructures, and requirements of accuracy and energy efficiency.

Energy-efficient Trajectory Tracking

Moves from ProtoGeo Oy



Compare Moves to other products

	Moves	Gadgets Nike+ Fuelband, Fitbit, etc.	Sports tracking apps Nike+ Running, Runkeeper, Runtastic, etc.
Automatic recognition of walking, bicycling and running	✓		
Calorie counter	✓*	✓	✓
Routes on map	✓		✓
Daily storyline with places	✓		
No need to start and stop	✓	✓	
No need to charge and carry an extra device	✓		✓

Case 1: Energy-efficient Trajectory Tracking

Mikkel Baun Kjærgaard, Sourav Bhattacharya, Henrik Blunck, and Petteri Nurmi. 2011. Energy-efficient trajectory tracking for mobile devices. In *Proceedings of the 9th international conference on Mobile systems, applications, and services* (MobiSys '11). ACM, New York, NY, USA, 307-320. DOI=10.1145/1999995.2000025 http://doi.acm.org/10.1145/1999995.2000025

Overview

- What to measure: current position and user's trajectory
- Sensors in use: GPS, compass, accelerometer
- Given a trajectory error threshold, the system will schedule sensor tasks(when and what to sense)

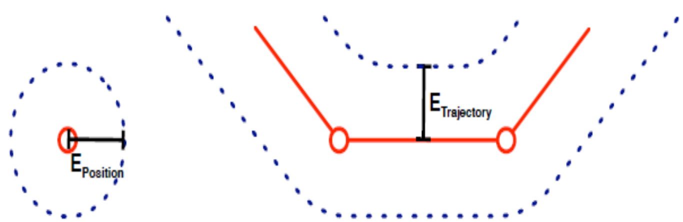
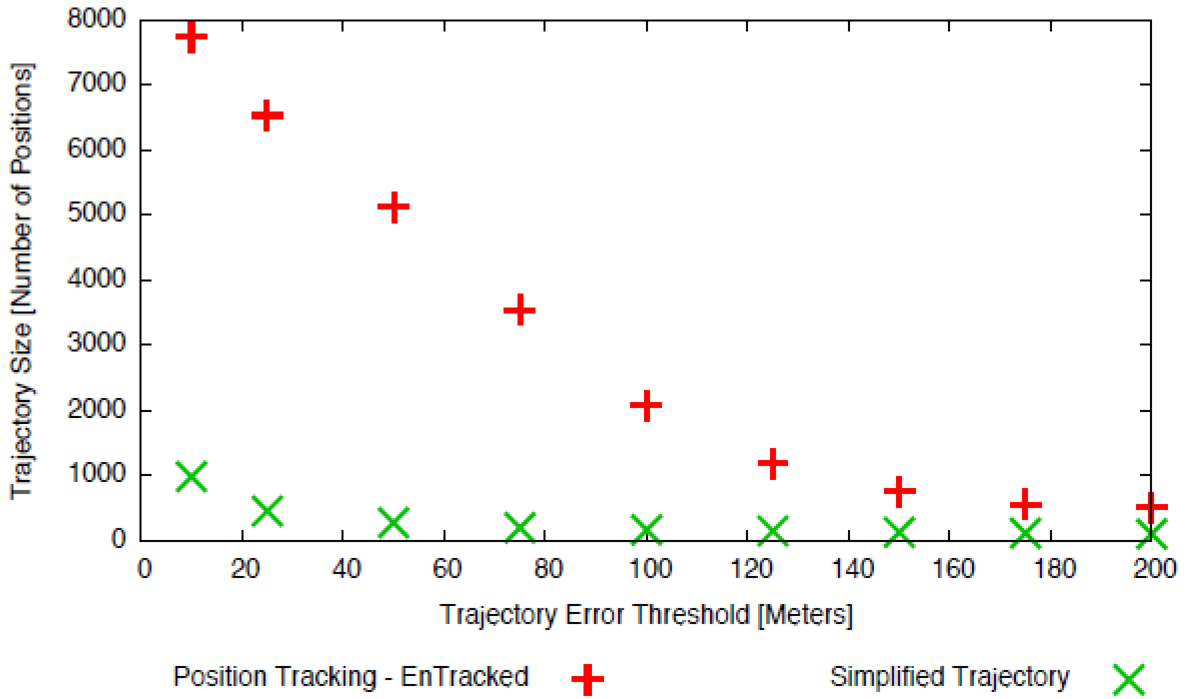


Figure 1: Illustrating error thresholds for position and trajectory tracking, respectively.

How many samples are enough?



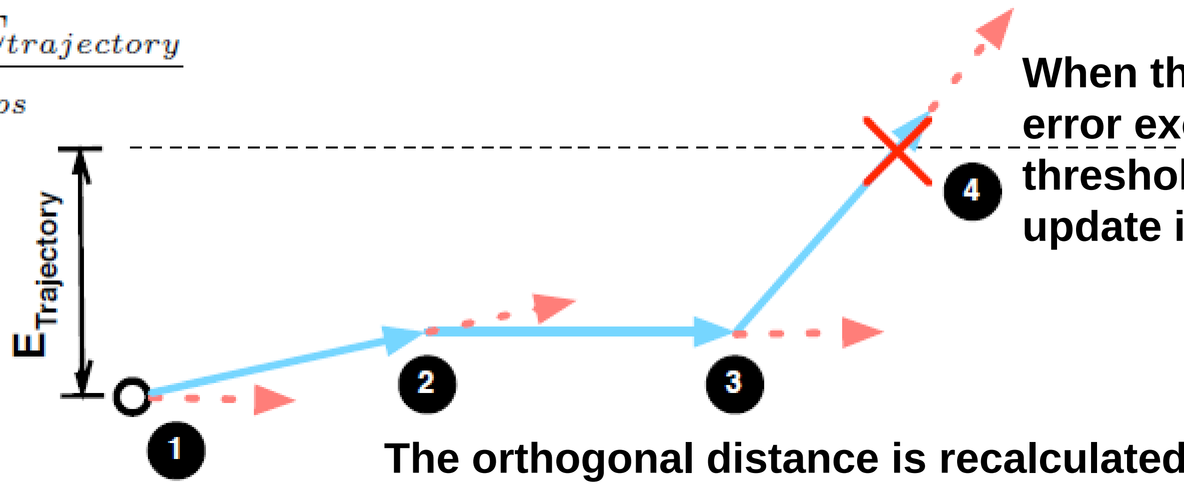
Try to minimize the usage of GPS for energy savings

Sensor Management Strategies

1) **Heading-aware Strategy: using compass as a turning point sensor**

✓ No explicit update of a target's position and trajectory is needed as long as the target is moving in a straight line with constant speed

$$\Delta t = \frac{(1-u) \cdot E_{trajectory}}{s_{gps}}$$

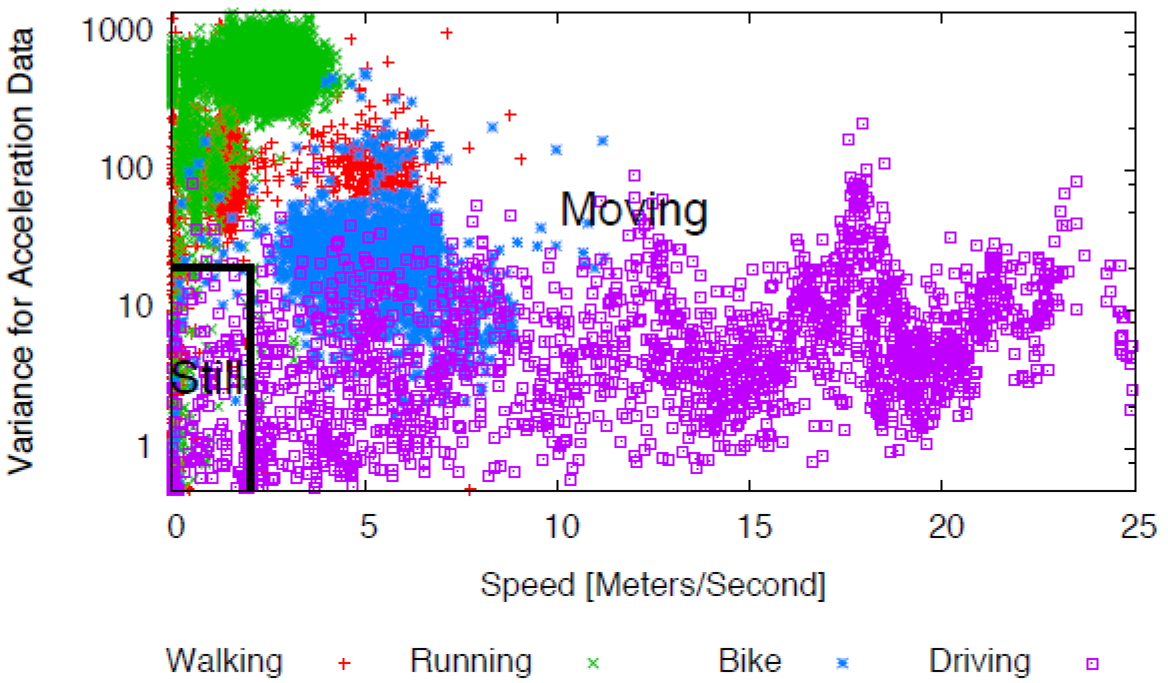


When the orthogonal error exceeds the error threshold, a new GPS update is needed

The orthogonal distance is recalculated using the previous GPS speed reading and the difference between the two most recent heading measurements

Sensor Management Strategies

2) **Movement-aware Strategy: using accelerometer and speed thresholds to detect stationary mode**

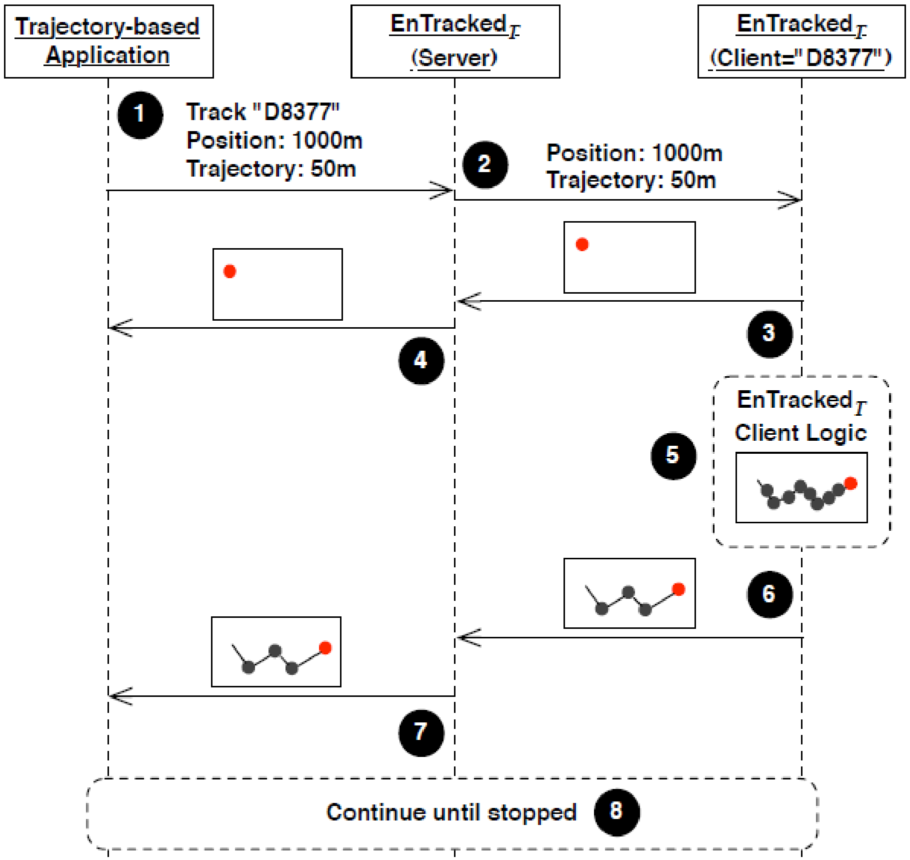


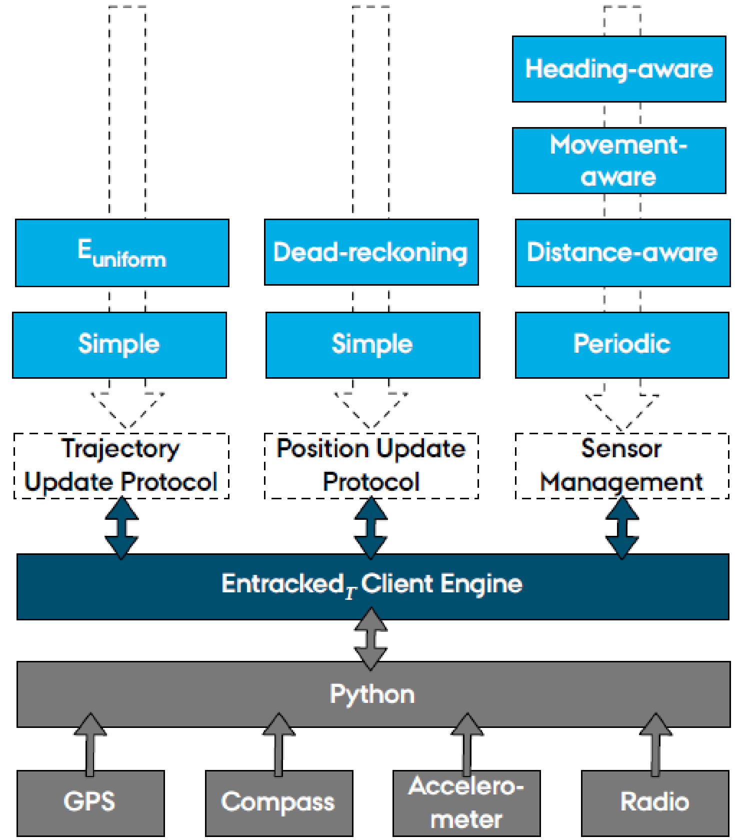
Sensor Management Strategies

3) Dynamically determining how long the GPS can sleep between successive position measurements

- ✓ Usually applied when the target is moving with a low speed and the trajectory error threshold is high, e.g., above 100 meters

System Architecture





Trajectory Simplification

- **Motivation:** reduce the energy cost for communicating trajectory data
- **Tradeoff:** reduced communication cost vs. increased computing cost (computing simplification)

- **Simplification:** select a subset of the points of the original polyline, so that the resulting simplified polyline does not deviate more from the original one than prescribed by a numeric error threshold

- Detailed algorithms can be found from the reference

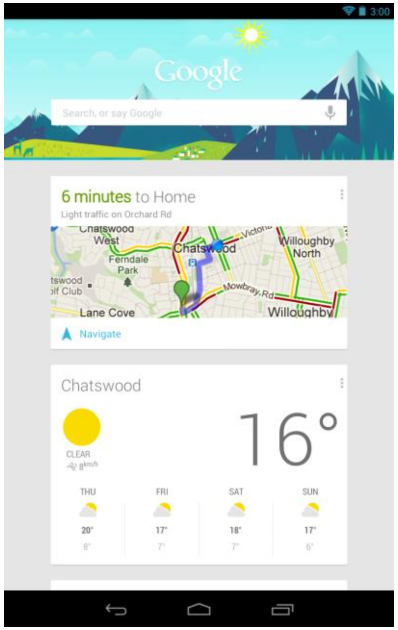
Evaluation

1) **Test case design**
   - Data sets were collected from walking, running, biking and car driving activities undertaken by different users
   - Varying the trajectory and positioning error thresholds used for tracking
2) **Power consumption is estimated based on power models of sensors(GPS, accelerometer, and compass), data transmission over 3G, and CPU**
3) **Accuracy of the tracked trajectories and positions vs. Energy consumption**

Case 2:

Yi Wang, Jialiu Lin, Murali Annavaram, Quinn A. Jacobson, Jason Hong, Bhaskar Krishnamachari, and Norman Sadeh. 2009. **A framework of energy efficient mobile sensing for automatic user state recognition.** In *Proceedings of the 7th international conference on Mobile systems, applications, and services* (MobiSys '09). ACM, New York, NY, USA, 179-192. DOI=10.1145/1555816.1555835 http://doi.acm.org/10.1145/1555816.1555835

User State Recognition is an Important Technique for Intelligent Assistant

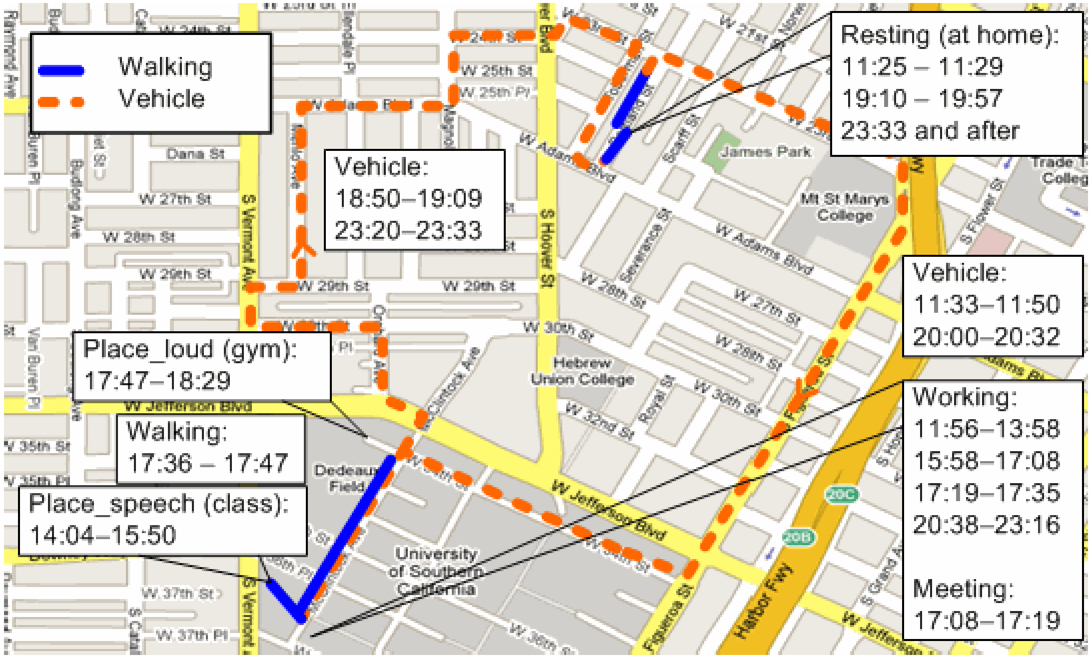


Overview

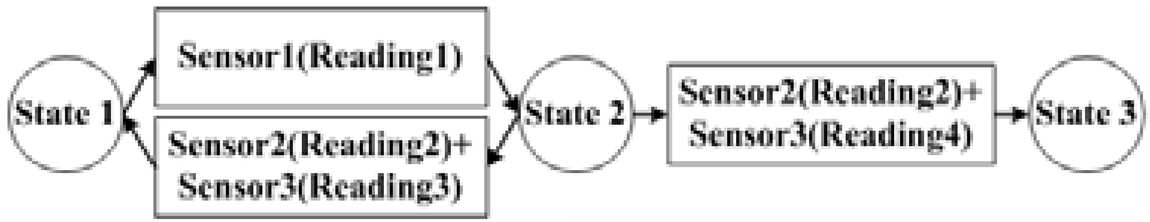
- **User state is described by 3 real-time conditions:**
  - ✓ Motion(such as running and walking)
  - ✓ Location(such as staying at home or on a freeway)
  - ✓ Background environment(such as loud or quiet)

- **Sensors used to recognize user state**
  - ✓ Accelerometer
  - ✓ Wi-Fi
  - ✓ GPS
  - ✓ Microphone

Example daily routines of a sample user

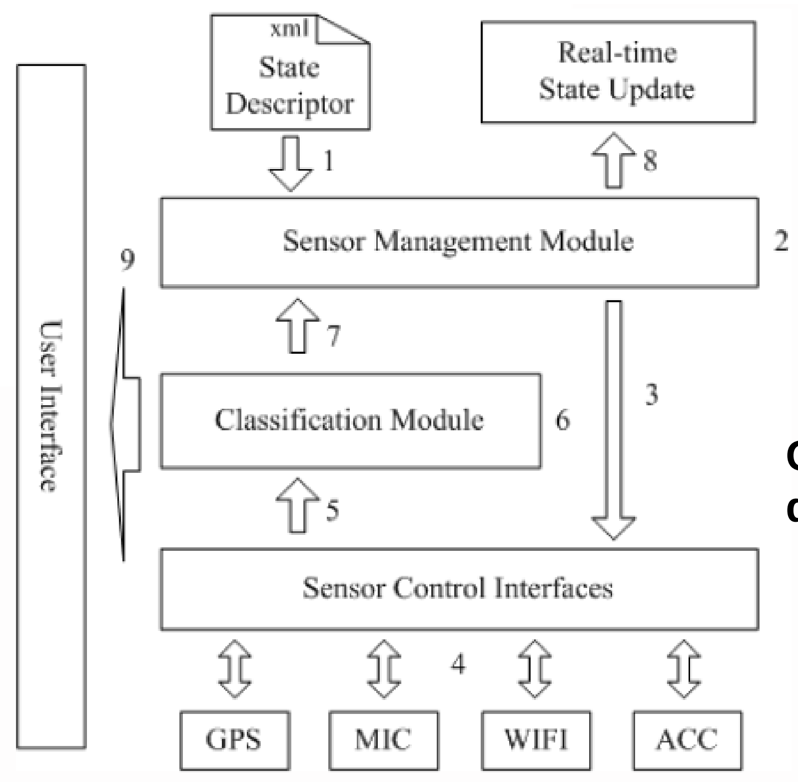


- **System Input: an XML-format state descriptor(including a set of state names, sensors to be monitored, and conditions for state transitions)**



- **A sensor management module is generated automatically based on the state descriptor**

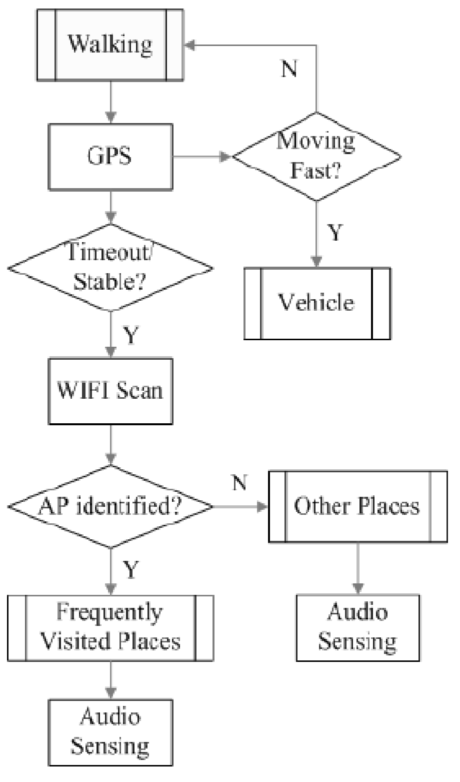
System Architecture



Management module instructs sensor control interface to turn on/off sensors

Classification module determines user state

Example: detect state transitions when the user is walking outdoor



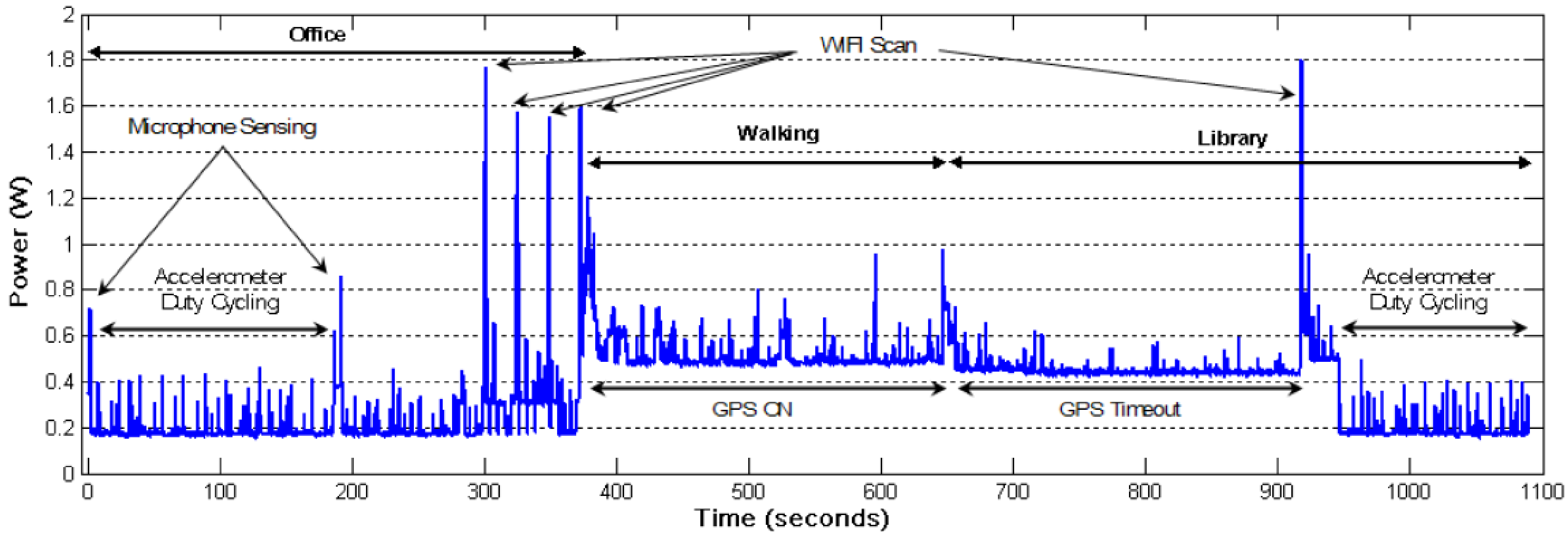
Once GPS times out due to lost of satellite singal or because the user has stopped moving for a certain amount of time, a WiFi scan will be performed.

The WiFi AP sets for one's frequently visited places such as home, office, etc. can be pre-stored on the device

Evaluation

- **Power consumption**
- **Accuracy of state recogition**
- **State transition detection latency**

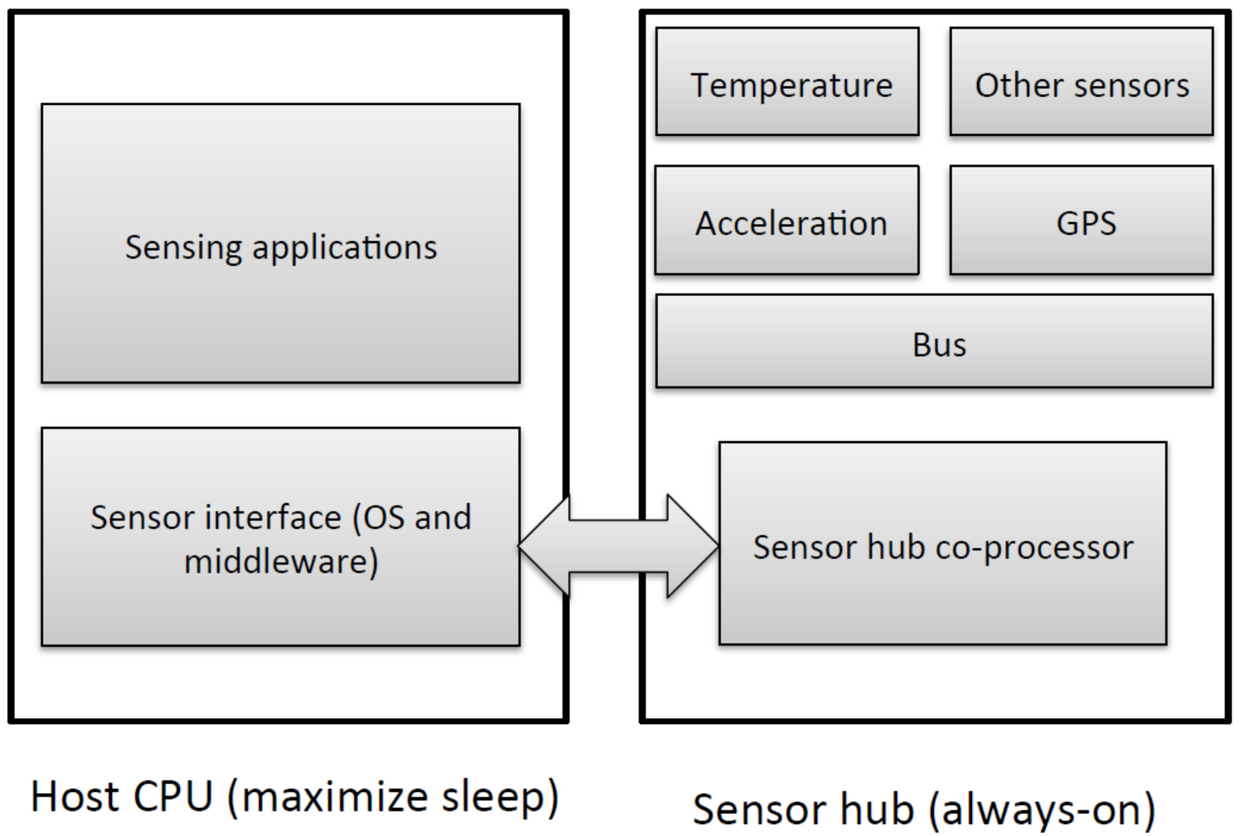
Power Consumption at a Glance



Stages of Sensor Data Processing

- **Sampling and buffering**, in which the sensors are sampled and the data is placed into a buffer.
- **Filtering**, in which the interesting parts of the data are identified and selected for further processing.
- **Feature extraction**, in which features are extracted from the data in order to perform classification.
- **Classification**, in which the data is classified based on the extracted features by using machine learning or probabilistic methods.
- **Post processing**, in which the applications react to the sensing result.

Sensor Hub


Sensing applications
Sensor interface (OS and middleware)
Temperature
Other sensors
Acceleration
GPS
Bus
Sensor hub co-processor
Host CPU (maximize sleep)
Sensor hub (always-on)

Examples of SensorHub Solutions

QuickLogic SensorHub
http://www.youtube.com/watch?v=Z01AGNkXSS8

Texas Instruments SensorHub

http://www.youtube.com/watch?v=saV77iUesCc

Atmel Corp.

http://www.youtube.com/watch?v=N6aYs80_boM

Questions?