21W.789 - Class 6
Mobile Programming
Mobile Location

Overview

- J2ME Programming
  - MIDlet Lifecycle
  - Differences from J2SE
  - Development Environment
  - Signing Code
- iPhone Programming
  - Development Environment
  - View Controllers
  - Application Lifecycle
- Mobile Location
  - GPS
  - Cell ID
  - Mixed Methods (Skyhook)
  - Location in J2ME/iPhone
  - Examples

J2ME – Familiar (sort of)

- At a high level, still java…

- Everything inherits from java.lang.Object

- Primitive types: int, float, boolean, byte, long, String

- Basic util/lang classes: Vector, Hashtable, Calendar, Date, Thread, Timer

- Basic i/o classes: InputStream, OutputStream, Reader, Writer, ByteArrayInputStream, ByteArrayOutputStream

- Classes exist in approximately Java 1.1.8 APIs

J2ME – What's Missing/Different

- Missing
  - Most classes from after Java 1.1.8…
  - Collections: Set, TreeSet, Properties
  - No built in XML support (can use KXmlParser)
  - AWT/Swing: J2ME has its own GUI classes
  - High level APIs: SQL, JNI, etc.
- Different
  - File I/O: standard File I/O classes are different
 - Networking: standard networking classes are different

JADs and JARs

- All code and resources must be packaged in a JAR file
- Metadata for the MIDlet goes into a JAD file
- Eclipse takes care of generating basic JAD (file size, main MIDlet, etc.) and JAR
- Example JAD:

MIDlet-Jar-URL: ZoneTag.jar

MIDlet-Jar-Size: 133285

MicroEdition-Configuration: CLDC-1.1

MIDlet-Version: 1.0.1

MIDlet-Name: ZoneTag

MIDlet-Vendor: Motorola

MicroEdition-Profile: MIDP-2.0

MIDlet-1: ZoneTag,,com.mot.labs.arc.zonetag.gui.ZoneTag

FlipInsensitive: True

MIDlet-Certificate-1-1:

MIIDojCCAoqgAwIBAgICCCwwDQYJKoZIhvcNAQEFBQAwfzELMAkGA1UEBhMCVVMxETAPBgNVBAgTCElsbGlub2lzMRUwEwYDVQQHEwxMaWJlcnR5dmlsbGUxFTATBg
NVBAoTDE1vdG9yb2xhIEluYzEMMAoGA1UECxMDUENTMSEwHwYDVQQDExhNYW51ZmFjdHVyZXIgRG9tYWluIDQwLTEwHhcNMDYxMTAzMDE0MTM5WhcNMjExMT
AzMDE0MTM5WjA
+MRUwEwYDVQQKEwxNb3Rvcm9sYSBJbmMxFTATBgNVBAsTDE1vdG9yb2xhIFBLSTEOMAwGA1UEAxMFQUM1NDEwgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoG
BAOxKOU3dXQgZqQzh++LbqVPi5n1fGPtKrL29xKejfwsOX5gX5rPDF//IyWZBs4ECrarptJvqZX8SLxmTwZUcQFz7y/2aOrF/FKYIzq/vIb/
Y2gbtpdeOjkDmV0dcIdpa8zqc5fjopKXt5U+3/
xbvyeX8d9aZUHFtGtpx28LpEd5zAgMBAAGjgewwgekwHwYDVR0jBBgwFoAUfHpHITdMaro5MMLC1vGdkWOiUy4wEQYJYIZIAYb4QgEBBAQDAgQQMA4GA1UdDwE
B/wQEAwIF4DATBgNVHSUEDDAKBggrBgEFBQcDAzB/BghghkgBhvkBFQEB/
wRwBo0ZYAgUCRZK1AAAAABV4AtQIhQJGxmFUEm2RCIijMwMHMx0AEAAiABgliKMLCwMnHQA0AAgAOAnZgYREQgHFq1yAAAAAGZgDTAZBwkbBChQQbZEIpwMLE
xsdIwAEACQAHAzrgbAEwYJJIPaAAAAADANBghghkgBhvkBFgQBATANBgkqhkiG9w0BAQUFAAOCAQEAI/iVIReYWELOHhRZqZBI8nv4hhkt0/
xThydmlaX9Rp0scRDLr2DOGYng0bgNGPIijw1JZFW/o3BUKTELK7+8YHwEURVitrSGIUFY/cQ8Qee+fgJJZBpJrfwJVmYEFfSRDUodRBypI
+73Dj0PwYJVI5xWs2vqM6qSLK4c9xwwn1rwYJxs9aXKvrxO4ftqelEZ8XYIMKpi+I7r02hBG9MMSpeCbbpe+1eSxF2DVXi/g/okY9lcVoxhko6paUZ3gZHTNEzAIPSU50/
mGXRGI2ELdk2MOTPji04e/by1PKD8agJB/RpD3NaLzrcZsKxAGgGuoXqIuMVRbgNFZoXa1IOh+A==

MIDlet-Jar-RSA-SHA1: nhVNq8gI9K4vqr69YWOZXuz2mVtHOd7n3cOdeSXxd6XR0AMgTE3THLOVqZmQ+hAskdQ26/
TIejyXHp6FrxrSRS7UICFuoq8Ax6jaVqWBuo0WSR10hVFqDfvWXVrQO0I6Do0ZSYaPIFAXWOJzC0uUicIFDRBKdQd3WR2vSD/fv3s=

MIDlet-Permissions: javax.microedition.io.Connector.http,javax.microedition.io.Connector.file.read,javax.microedition.io.Connector.file.write,javax.microedition.io.Connector.socket

MIDlet-Permissions-Opt: javax.microedition.media.control.VideoControl.getSnapshot

MIDlet Lifecycle

Constructor – gets called once, when MIDlet is first created

startApp() – get called the first time the MIDlet is invoked and everytime the MIDlet resumes from a suspended state

pauseApp() – is called if the user changes focus to another phone application or phone event (e.g. phone call) requires the MIDlet to be suspended

destroyApp(boolean arg0) – is called if the user chooses to exit the application (e.g. presses red home button) or if the phone needs more resources for another process. You can throw an exception to say you'd really not like to quit, but the platform can override
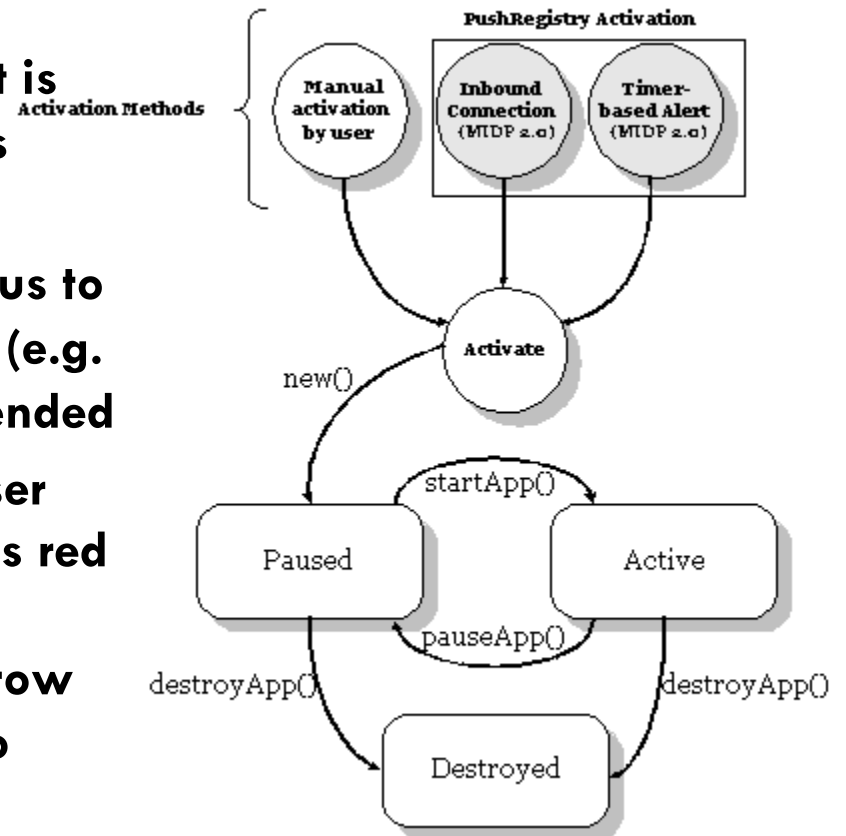
MIDlet Permissions

- Whenever you use a method that is protected, you must declare it in the permissions of the JAD file – example:
  ```
  MIDlet-Permissions:
  javax.microedition.io.Connector.http,javax.microedition.io.
  Connector.file.read,javax.microedition.io.Connector.file.wr
  ite,javax.microedition.io.Connector.socket
  ```

- Optional permissions allow application to run on phones that may not have all APIs - example:
  ```
  MIDlet-Permissions-Opt:
  javax.microedition.media.control.VideoControl.getSnapshot
  ```

- If you don't declare a permission and try to use a protected API, likely a SecurityException will be thrown, or the KVM will just exit

Example Permissions

□ Common Permissions:
- javax.microedition.io.Connector.http
- javax.microedition.io.Connector.https
- javax.microedition.io.Connector.datagram
- javax.microedition.io.Connector.datagramreceiver
- javax.microedition.io.Connector.socket
- javax.microedition.io.Connector.serversocket
- javax.microedition.io.Connector.ssl
- javax.microedition.io.Connector.comm
- javax.microedition.io.PushRegistry
- javax.wireless.messaging.sms.send
- javax.wireless.messaging.sms.receive
- javax.microedition.io.Connector.sms
- javax.wireless.messaging.cbs.receive
- javax.microedition.media.control.RecordControl.record
- javax.microedition.media.control.VideoControl.getSnapshot
- javax.microedition.pim.ContactList.read
- javax.microedition.pim.ContactList.write
- javax.bluetooth

J2ME Connector API

- All stream I/O is initiated by the Connector class
- Connector can get you HTTP streams, File streams, Sockets, SMS, etc.

- HTTP Example:

```
HttpConnection c =
      (HttpConnection)Connector.open("http://web.mit.edu/index.html");
InputStream is = c.openInputStream();

// read from input stream here

is.close();
c.close();
```

J2ME Connector (files)

- File Example:
  ```
  FileConnection sc =
  (FileConnection)Connector.open("file:///c/mobile/
  picture/tmp.txt");
  OutputStream os = sc.openOutputStream();
  os.write(("text to go into the file").getBytes());
  os.flush();
  os.close();
  ```

- Don't forget to add the appropriate permissions to your JAD file and sign if necessary (i.e. when using files)!!

RMS – persistent storage

- RMS – the Record Management Store – is an easy place to store persistent data

- Can create multiple stores that each contain a set of records

- Example adding a record:

```
RecordStore rs =
RecordStore.openRecordStore("MyAppointments",true);
String appt = "new record";
byte bytes[] = appt.getBytes();

rs.addRecord(bytes,0,bytes.length);
rs.closeRecordStore();
```

- Example reading a record:

```
RecordEnumeration re = rs.enumerateRecords(null, null, false);
if (re.hasNextElement())
 byte nextRec[] = re.nextRecord();
```

- More info: http://www-128.ibm.com/developerworks/library/wi-rms/

Messaging in J2ME

- You can send and receive SMS messages from Java. To talk between applications, you can address an SMS to a specific port that an application on another phone can listen to
- Can send/receive both text and binary messages (limited to 160 bytes)
- Example:

sender = (MessageConnection)Connector.open("sms://+16172531000:9532 ");

TextMessage t =
        (TextMessage)sender.newMessage(MessageConnection.TEXT_MESSAGE);

t.setPayloadText("Hello World");

t.setAddress("sms://+16172531000:9532");

sender.send(t);

- Server:

serverConn = (MessageConnection)Connector.open ("sms://:9532");

serverConn.setMessageListener(this); // where this implements MessageListener

Push Registry

- The push registry is a way to automatically start a MIDlet after a specified amount of time or when a given system event occurs (e.g. incoming SMS)

- You can register statically in the JAD, or dynamically in the constructor of your MIDlet…

- Example:

   MIDlet-Push-1: socket://:5000, j2medeveloper.basicpush.PushMIDlet, *

   MIDlet-Permissions: javax.microedition.io.PushRegistry, javax.microedition.io.Connector.serversocket

- More info: http://developers.sun.com/techtopics/mobility/midp/articles/pushreg/

J2ME UI

- Two different ways to build a screen in J2ME:
  - LCDUI
    - Standard widgets (lists, checkboxes, text fields, etc.)
    - Look like standard platform components on any phone platform
    - Easy to implement
    - Doesn't allow access to number pad except when entering text
  - Game Canvas
    - Can implement your own widgets
    - Full control of screen and keypad
    - More difficult to customize (e.g. what happens to a widget on a different screen or a phone with different keymaps)

LCDUI

- LCD UIs are a series of screens
- Common interactions are wizard-like with a series of lists as the interaction
- Forms can contain a series of items on one screen
- Can intermix LCD UI screens and canvas-based screens in an application
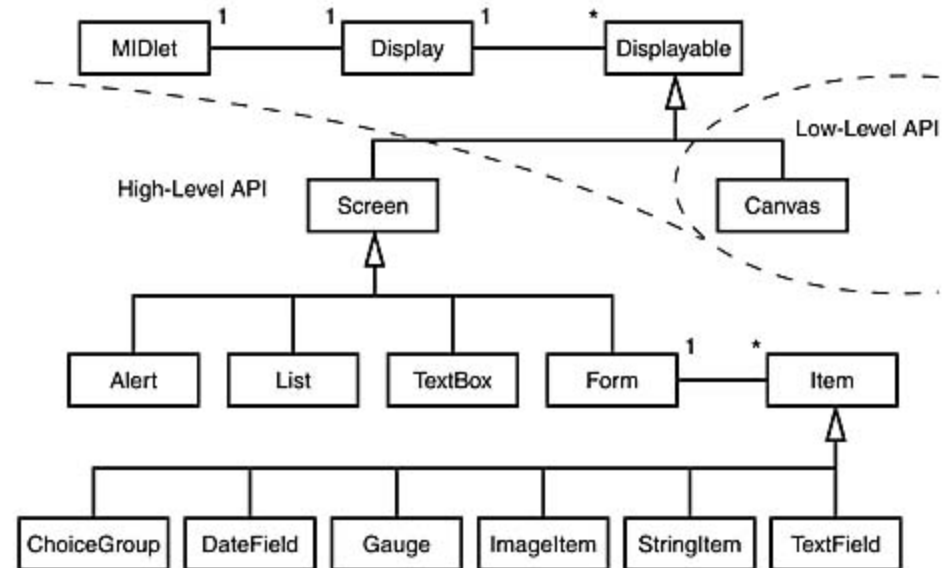
Security / Signing

- Some APIs are protected such that only applications signed by the manufacturer or carrier can use them
- Examples:
  - Cell ID
  - File I/O to parts of the phone file system
  - Capturing a full-resolution image from the camera

- Follow instructions in the Motorola certificate signing guide to use openssl to generate the SHA1 hash of your Jar file and add MIDlet-Certificate-1-1 and MIDlet-Jar-RSA-SHA1 to your JAD file before loading it onto the phone

iPhone Development

- Need to have a paid Apple Developer account
- Generate certificate with UDIDs of devices

- Distribution
  - Debug load directly on phone
  - AdHoc distribution (.mobileprovision file + app bundle)
  - iTunes store

Objective C

- Superset of C
    - Can Mix C/C++ and Objective C
    - Single Inheritance
    - Loosely typed (treat compiler warnings seriously!)
- Syntax:
    - [instance method];
    - [instance method:arg1 arg2name:arg2];

Strings, Logs, and Arrays

- Strings
  - NSString *myString = @"my string";
  - [NSString stringWithFormat:@"with number: %d",5];
- Logging
  - NSLog(@"debug info here");
- Arrays
  - NSArray *array = [NSArray arrayWithObjects:@"One", @"Two", @"Three", nil];
  - If any of your objects is nil, array will not be full!!

View Controllers

- Application contains a NavigationController

- Each screen is a UIViewController

- New screens appear with a push of a View Controller onto Navigation Controller:

  [[self navigationController] pushViewController:targetViewController animated:YES];

Application Lifecycle

- Only one (non-Apple) application can be running at a time

- Applications suspended when phone sleeps or when interrupted (e.g. incoming call)
  - On wake-up, -(void)applicationDidBecomeActive called on AppDelegate
  - All state maintained, but no execution occurs while application is inactive

iPhone resources

- http://www.stanford.edu/class/cs193p/cgi-bin/index.php Stanford iPhone Class
- http://ericasadun.com/ Erica Sadun's iPhone Coo kbook
- http://www.cocoabuilder.com/archive/bydate Cocoabuilder
- http://cocoadevcentral.com/articles/000082.php CocoaDevCentral: Cocoa Style for Objective-C: Part I
- http://www.iphonesdkarticles.com/ iPhone SDK Articles
- http://cocoadevcentral.com/ Cocoa Dev Central
- http://icodeblog.com/ iCodeBlog
- http://theocacao.com/document.page/510 Theocacao
- http://www.v2ex.com/tag/uitableviewcell/ UITableViewCell | V2EX
- http://idevkit.com/forums/tutorials-code-samples-sdk/30-custom-uitableviewcell.html Custom UITableViewCell - iDevKit
- http://peagolon.wordpress.com/2008/11/15/using-uitableviewcell-with-interfacebuilder/ Building UITableViewCell with IB
- http://discussions.apple.com/thread.jspa?threadID=3D1579070&tstart=3D43 Loading views in landscape orientation
- http://discussions.apple.com/thread.jspa?threadID=3D1603141&tstart=3D27 Half-curl transitions
- http://cocoawithlove.com/2008/12/heterogeneous-cells-in.html Heterogeneous cells in a UITableViewController
- https://www.nearinfinity.com/blogs/scott_leberknight/iphone_bootcamp_blogs.html Phone bootcamp blogs
- http://www.sleberknight.com/blog/sleberkn/entry/iphone_bootcamp_day_4 iPhone bootcamp blogs 2
- http://www.iphonedevsdk.com/forum/iphone-sdk-development/4879-uitableview-cell-deletion-methods.html UITableViewCell deletion methods
- http://savoysoftware.com/blog/ enhancing performance iPhone
- http://stackoverflow.com/questions/328391/last-indexed-cell-in-uitableview-is-taking-on-wrong-font Cell Identifiers
- http://stackoverflow.com/questions/tagged/iphone StackOverflow
- http://www.cocoadev.com/index.pl?NSUserDefaults NSUserDefaults
- http://knol.google.com/k/usman-ismail/iphone-sdk-application-preferences/34oprzanmpe7q/8# Application Preferences tutorial
- http://icodeblog.com/2009/02/02/great-resource-for-all-iphone-developers-ibetatestcom/ iBetaText.com
- http://blog.coriolis.ch/2008/11/09/add-an-uiprogressview-or-uiactivityindicatorview-to-your-uialertview/ progressView
- http://idevkit.com/forums/general-sdk/299-nsurlconnection-nshttpcookie.html NSURLConnection, NSHTTPCookie - iDevKit
- http://stackoverflow.com/questions/576265/convert-nsdate-to-nsstring Convert NSDate to NSString - Stack Overflow
- http://www.cocoadev.com/index.pl?DescriptionWithCalendarFormat CocoaDev: DescriptionWithCalendarFormat
- http://www.planetcocoa.org/ Planet Cocoa
- http://www.oiledmachine.com/posts/2009/01/04/managing-concurrent-asynchronous-url-requests-in-cocoa.html Managing concurrent asynchronous URL requests in Cocoa

Mobile Location

- GPS
- Cell ID
- Mixed Methods (Skyhook)
- Location in J2ME
- Location on iPhone
- Example Applications
  - ZoneTag
  - Motion Presence
  - Place Its
  - Jogging the Distance
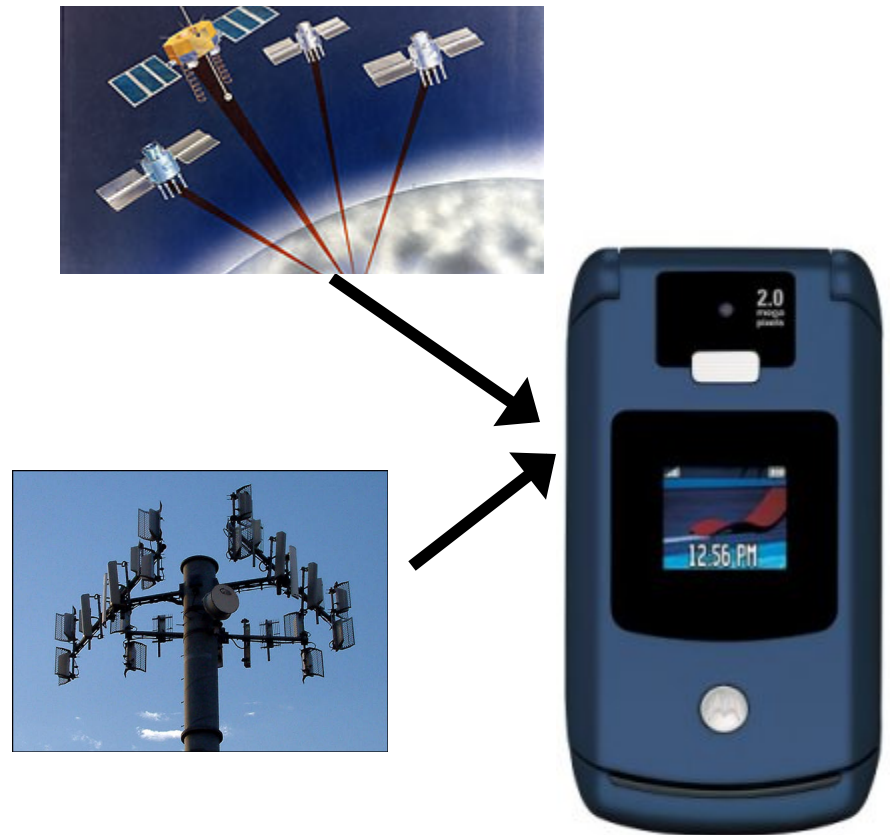
Cell ID

- Use the cell infrastructure to know when you are in a given location
- In GSM networks, all cells in the world have a globally unique ID made up of four numbers:
  - cell id, lac, mnc, and mmc
- Databases are being made that map cell IDs into locations
- Cells can be manually mapped to semantic locations (e.g. Campus, Home, Downtown Boston, etc.)

Cell Topology

Downtown | Residential (cropped)
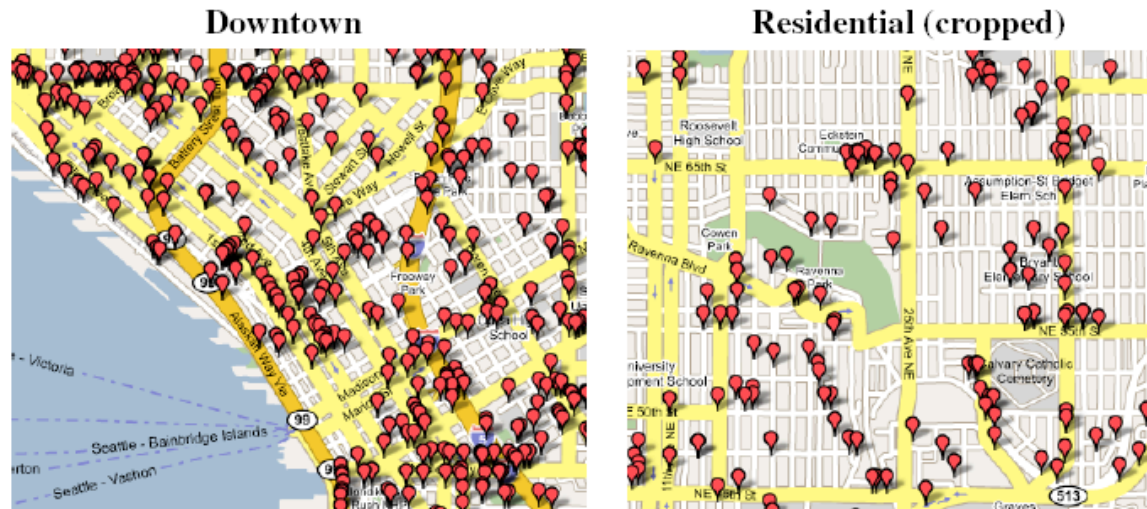
Fig. 3. Cell location map for the three network providers; each dot represents the estimated location of a cell. The left map shows Downtown with an average density of 66 cells/Km^2. The right map shows a cropped Residential region with an average cell density of 26 cells/Km^2.

(from Chen et al 2006, http://www.intel-research.net/Publications/Seattle/100920061625_366.pdf)

Much greater density in cities where places are also more dense

Cell ID -> Location (update this)

- Databases exist that are trying to map a large number of Cell IDs to locations

- Place Lab
  - Intel project mapping WiFi and GSM Cell ID to GPS coordinates
- ZoneTag
  - Yahoo! Research Berkeley project mapping GSM Cell ID to Zip Code

- You can contribute data you collect in this class to either database

Cell ID - Applications

- People have used Cell ID to:
  - Determine when someone was at a given place (e.g. home) (see Sohn et al '05)
  - Distinguish between walking/driving/stationary (see Sohn et al '06)
  - Provide accurate positioning (e.g. Chen et al '06)
    - Down to 95% error of 163m with all visible cells

GPS

- 30 satellites in earth orbit
- 6 satellites always in line of site of any place on earth

- Receivers must find 4 or more satellites for a fix

- Receiver listens for time broadcasts from each satellite and calculates distance based on time delay
- Satellites also broadcast Navigation Messages which can be used with the time delays to calculate position
- Receivers often provide access to position data as a NMEA stream (National Marine Electronics Association)

GPS – NMEA

- A compound string from the NR203 GPS Receiver containing multiple messages. The actual messages decoded include…

- ZDA – Standard NMEA $..ZDA Time & Date message
- GLL – Standard NMEA $..GLL Geographic Position – Latitude/Longitude message
- NSV – NMEA message containing individual satellite information.
- Typical ASCII String
  $<CR><LF>
  MRK,0<CR><LF>
  ZDA,123336.8069,17,06,2001,13.0<CR><LF>
  GLL,2924.11158,N,1211.07392,W, 75.97,M<CR><LF>
  VTG,218.7,T,2.38,H,0.18,V<CR><LF>
  SGD,-1.0,G,-1.0,M<CR><LF>
  SYS,3T,9<CR><LF>
  ZEV,0.28745E-006<CR><LF>
  NSV,2,00,000,00,0.0,00.0,00,00,D<CR><LF>
  NSV,7,00,000,00,0.0,00.0,00,00,D<CR><LF>
  NSV,28,00,000,00,0.0,00.0,00,00,N<CR><LF>
  NSV,1,00,000,00,0.0,00.0,00,00,D<CR><LF>
  NSV,13,00,000,00,0.0,00.0,00,00,D<CR><LF>
  NSV,4,00,000,00,0.0,00.0,00,00,N<CR><LF>
  NSV,25,00,000,00,0.0,00.0,00,00,N<CR><LF>
  NSV,0,00,000,00,0.0,00.0,00,00,N<CR><LF>
  NSV,11,00,000,00,0.0,00.0,00,00,D<CR><LF>
  NSV,0,00,000,00,0.0,00.0,00,00,N<CR><LF>
  &

- (from http://en.wikipedia.org/wiki/NMEA_0183)

GPS on the Phone

- Some phones have built in GPS
  - iPhone, G1, Motorola Q
  - All Motorola phones on the iDEN (Sprint/Nextel) network
  - Many CDMA phones (e.g. RAZR on Verizon)

- Phones that don't have built-in GPS can use a bluetooth unit
  - sends a NMEA stream to the phone
  - phone can listen through a BT serial connection

Alternatives to GPS and Cell ID

- Bluetooth Beacons
  - Known bluetooth devices associated with a given locations

- WiFi positioning
  - Known wifi SSIDs mapped to locations

- Bluetooth proximity
  - Around Ed, Kevin, students = in class

- RFID
  - Tags associated with places, reader in handset

Wifi Positioning

- Use available SSIDs to determine location

- Up to 40m accuracy in cities

- Issues:

  - Need to have sampled fairly closely to where you want to get location

  - Not practical until recently when combined with other methods to bootstrap

Mixed Methods

- Phones have GPS, Wifi, Cell ID
- Use any available to determine location
    - Save other data to help improve location for others
- Provides "good-enough" positioning in seconds while waiting for GPS to attach and converge

- iPhone using Skyhook (100m requests/day)
- Yahoo using Navizon

Location in J2ME

- Cell ID

```
String cellID = System.getProperty("phone.cid");

String lac = System.getProperty("phone.lac");

String mcc = System.getProperty("phone.mcc");

String mnc = System.getProperty("phone.mnc");


cellID + lac + mcc + mnc = a globally unique ID for a
cell tower
```

Location on iPhone

- All location queries handled by CLLocation Manager
  - Uses a variety of means to determine location (Cell ID, Wifi positioning, AGPS, GPS)
  - Returns asynchronously as location is refined
  - Gives accuracy with each return
- locationManager = [[[CLLocationManager alloc] init] autorelease];
  locationManager.delegate = self;
  [locationManager startUpdatingLocation];
- // Called when the location is updated
  - (void)locationManager:(CLLocationManager *)managerdidUpdateToLocation:(CLLocation *)newLocation   fromLocation:(CLLocation *)oldLocation

Applications

- Location Centered Applications
  - ZoneTag
  - Motion Presence
  - Place Its
  - Jogging Over a Distance
- Location as an enhancement
  - Flixster
  - Yelp
  - Countless other iPhone/Android applications

ZoneTag (2006)

- Yahoo! Research project (J2ME app in collaboration with Motorola)
  - Find zip-code-level location using Cell ID
  - Use Yahoo! Local + social tag history to suggest tags to apply to photos
  - Upload to Flickr with tags and location
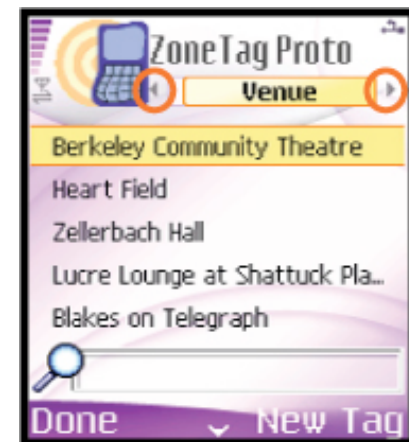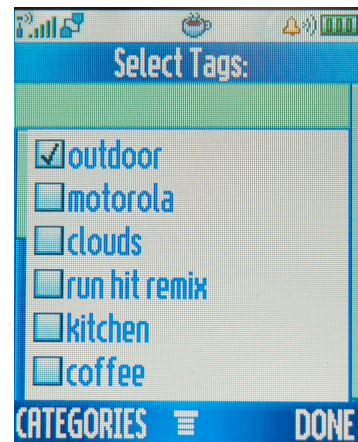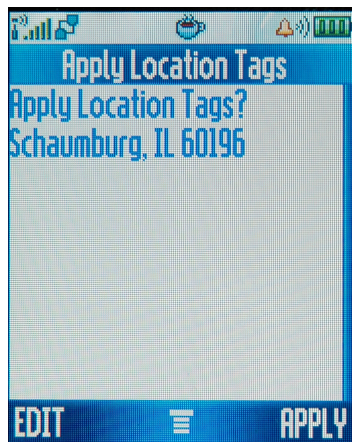
Motion Presence (2006)

- Inspiration:
  - Trying to identify transition times between places
- Research questions:
  - Will knowing if someone is in motion or at a place help people micro-coordinate?
  - Does not sharing the location itself mitigate privacy concerns?
  - What can be inferred from motion data?

Contacts

Frank / not moving (5)	
Crysta / moving (14)	
Gunnar / not moving (15)	
Larry / moving (5)	
Prashant / moving (25)	

Back   Select

Motion Presence Implementation

- Native Linux Process on phone
  - Monitored cell id transitions, determined when person was moving or "at a place"
  - Sent SMS message to a port on the devices of friends and family
  - Received SMS messages and maintained state of others
- J2ME Phonebook Application
  - Read status created by native process
  - Allowed for calling/texting
  - Showed user's own motion status on top line

Motion Presence – Field Study

- 10 users (4 social groups)
- Used application on primary phone for 2 weeks
- Nightly voicemail diary
- 2 interviews
- Application logging

Motion Presence - Data

- "George, I knew he was going to work, but I wasn't sure if he got there already and **I saw that he was not moving for 12 minutes. So judging by that I'm getting that he was already at work so I didn't bother calling him.**" (Harold)

- "If you knew someone was going to go pick you up or if someone was going to go someplace and you knew that and you know about what time, **you could see if they were actually on their way or if they were running late. …** Kind of lets you know when you should be ready or things like that." (James)

- "I've been working a lot and I'm not with Ebony, **so I've been looking at it just to see the motion on the phone.**" (Farisa)

Motion Presence – Findings

- Participants were able to infer: Location, Activity, Availability, Arrival Time, Destination
- Participants used the application to:
  - Moderate availability
  - Arrive at the same time
  - Get more time at their current activity
  - See people were following through on commitments
  - Check on other's safety
  - Social awareness – know what's going on with others
- Privacy not a major concern given plausible deniability

Place Its (2006)

- The problem: getting reminders in particular locations (e.g. call my mom when I get home, stop at the grocery store on the way home, etc.)
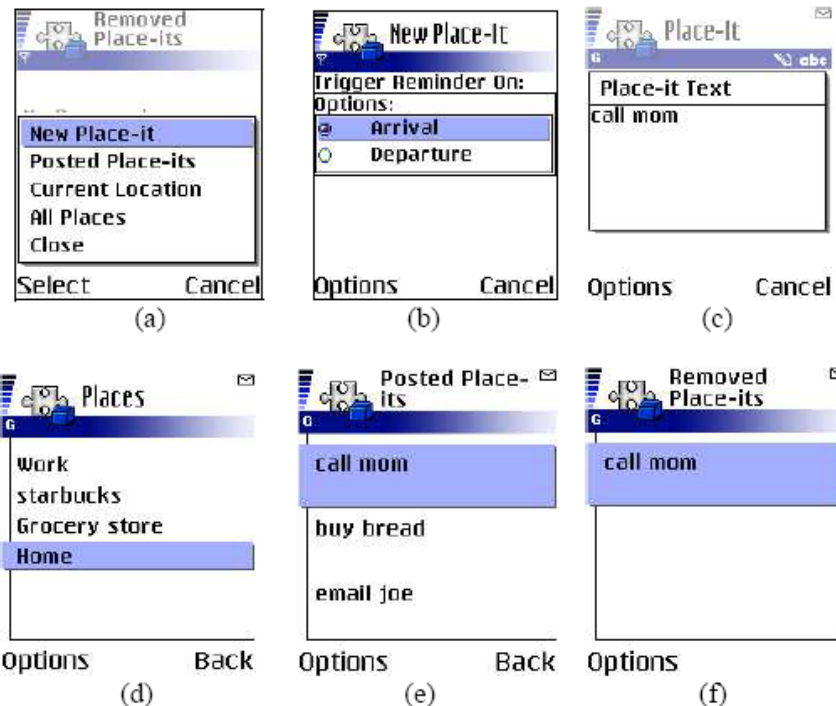

Figure 1. (a) Creating a new Place-It note; (b) Setting the note to be triggered upon arrival; (c) Typing the text of the note; (d) Posting the note to 'Home'; (e) Showing all posted Place-It notes; (f) The reminder is triggered when Jill arrives at the home and the note is removed.

Jogging the Distance (2007)

- Uses GPS to determine who is running faster

- Changes audio to sound like you're ahead or behind your running partner

- Demonstrated at CHI 2007

Partner is jogging faster

Jogging at same speed

Partner is jogging slower
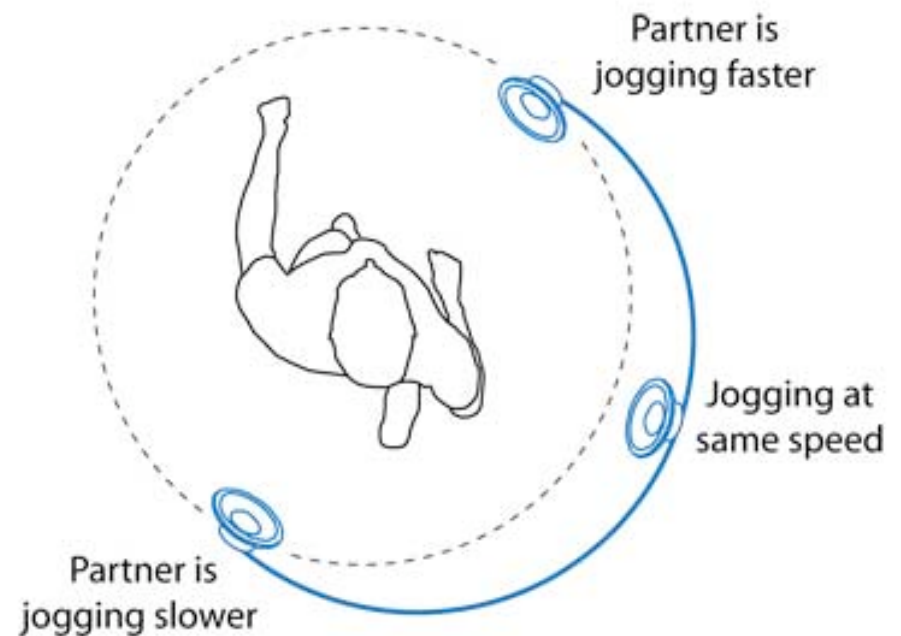
Jogging the Distance

Location as UI Enhancement

- New applications where location is meant to improve interaction, not be the central component
  - Make search/navigation easier (Yelp for iPhone)
  - Jump to relevant information (Flixster)

Assignment – In Project Teams

- On a phone that supports Cell ID
    - Count the Cell IDs visible on campus
    - Produce a map with places where particular IDs are visible
    - What can you say about cell topology around campus?
- On phones that support GPS
    - Create application that gets GPS coordinates and error
    - Where is error greatest?
    - How does indoor performance compare to outdoor?
    - Are some buildings different than others?
    - What might be causing these differences?
- Prepare a 5 minute presentation of your results (email to Ed, Frank, Kevin before class)