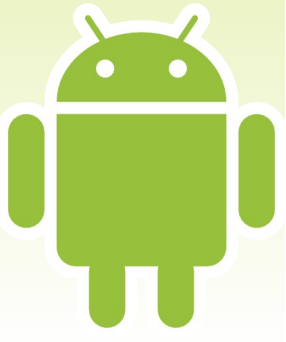
Advanced Maps V2

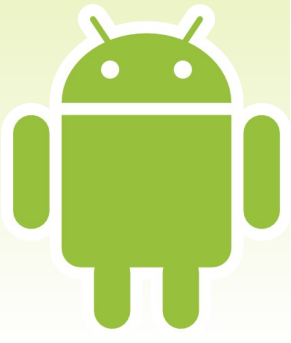


What You Should Already Know

- Maps V1 vs. Maps V2
- Device Requirements
- Play Services SDK and API Key
- Manifest Settings
- MapFragment, SupportMapFragment, or MapView
- GoogleMap

A Warning About IPC

- Most interactions with GoogleMap involve IPC
  - Map is rendered by separate process
- Most interactions with GoogleMap must be done on main application thread
  - For no obviously good reason
- Net: be careful about performance!
  - Allow main application thread to “breathe” if you are doing lots and lots of calls



Marking Up a Map

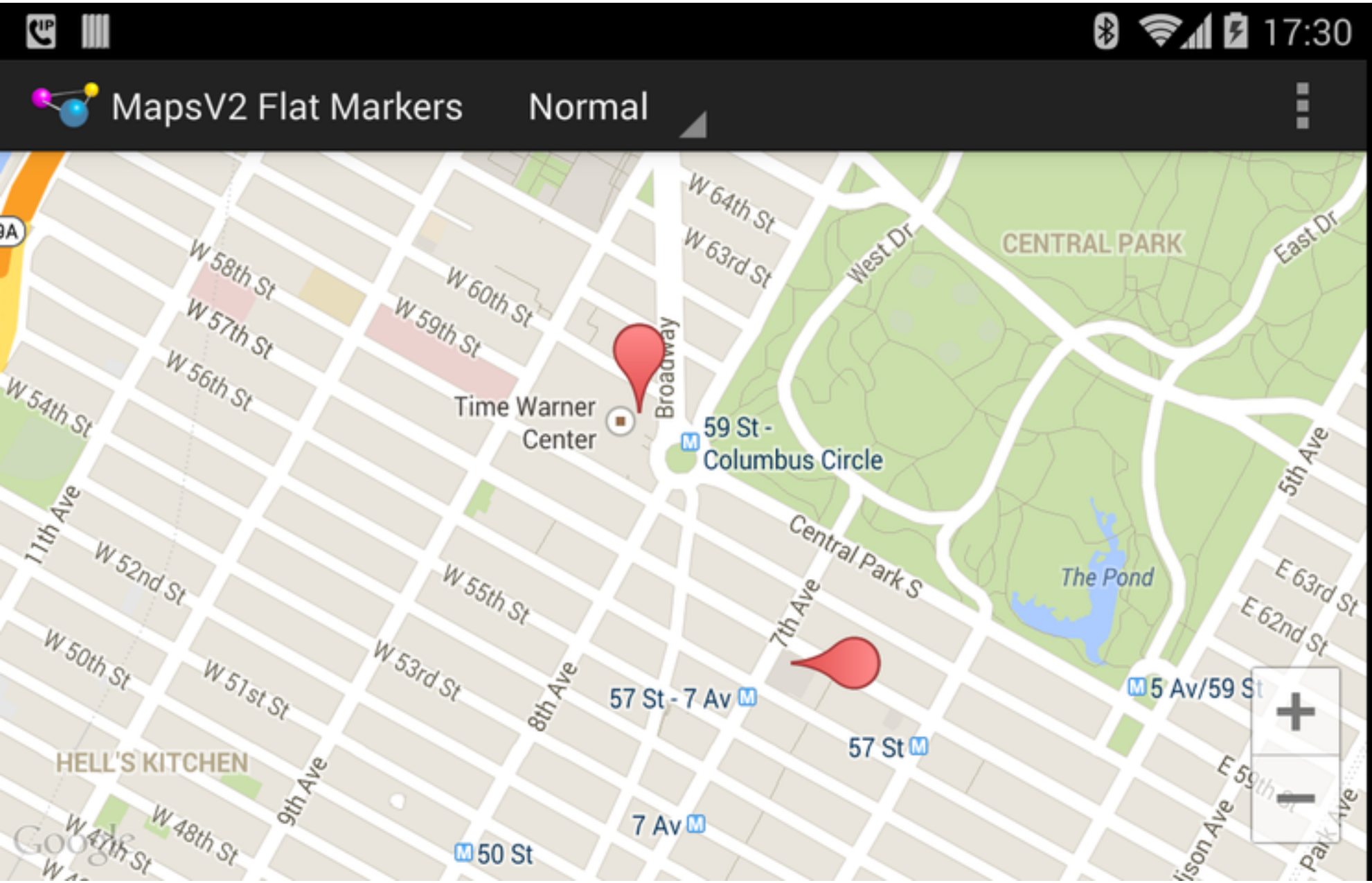
- No More Overlays!
- Add Markers Via `addMarker()`
  - Takes a MarkerOptions object
  - Fluent API to describe marker
  - Position as a LatLng
    - No more microdegrees!
  - Provide title and snippet for default pop-up "info window"

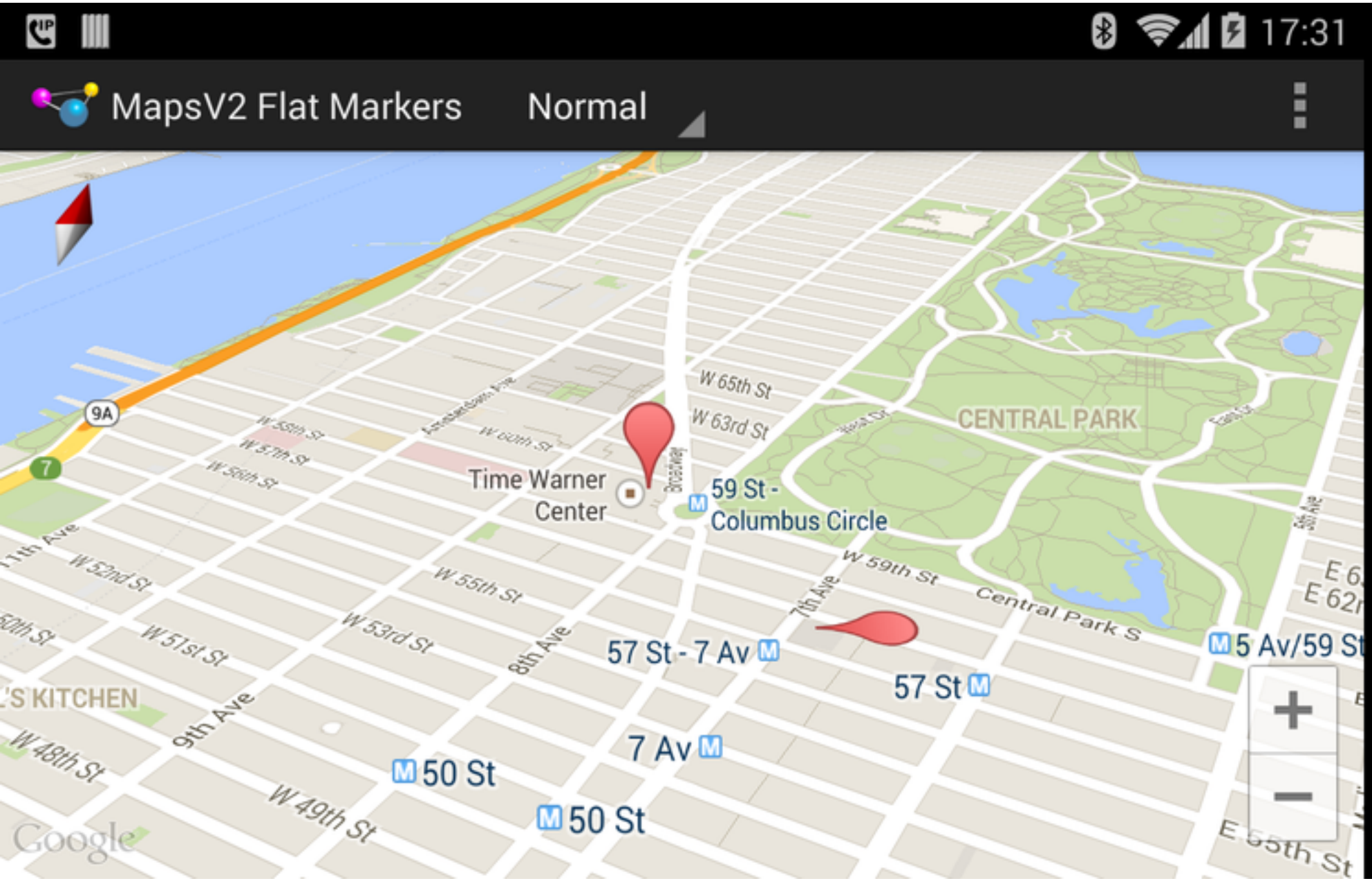
Marking Up a Map

- Marker Icons
  - Default
  - Yours
    - icon() method on MarkerOptions, taking a BitmapDescriptor from a BitmapDescriptorFactory
  - Flat
    - Drawn on map surface, rather than “pushed in”
  - Rotated
    - Useful with flat markers, to “point” to particular direction

```
addMarker(map, 40.748963847316034, -73.96807193756104,
          R.string.un, R.string.united_nations, false, 180);
addMarker(map, 40.76866299974387, -73.98268461227417,
          R.string.lincoln_center,
          R.string.lincoln_center_snippet, false, 0);
addMarker(map, 40.765136435316755, -73.97989511489868,
          R.string.carnegie_hall, R.string.practice_x3, true, 90);
addMarker(map, 40.70686417491799, -74.01572942733765,
          R.string.downtown_club, R.string.heisman_trophy, true,
          270);
```

```
private void addMarker(GoogleMap map, double lat, double lon,
                       int title, int snippet, boolean flat,
                       float rotation) {
  map.addMarker(new MarkerOptions().position(new LatLng(lat, lon))
                                   .title(getString(title))
                                   .snippet(getString(snippet))
                                   .flat(flat).rotation(rotation));
}
```


17:30
MapsV2 Flat Markers
Normal
W 64th St
W 63rd St
West Dr
CENTRAL PARK
East Dr
W 58th St
W 57th St
W 60th St
W 59th St
W 56th St
W 54th St
Broadway
Time Warner Center
59 St - Columbus Circle
5th Ave
11th Ave
W 52nd St
Central Park S
The Pond
E 63rd St
E 62nd St
W 55th St
7th Ave
W 50th St
W 51st St
W 53rd St
8th Ave
57 St - 7 Av
5 Av/59 St
57 St
HELL'S KITCHEN
9th Ave
E 59th St
W 48th St
7 Av
50 St
Park Ave

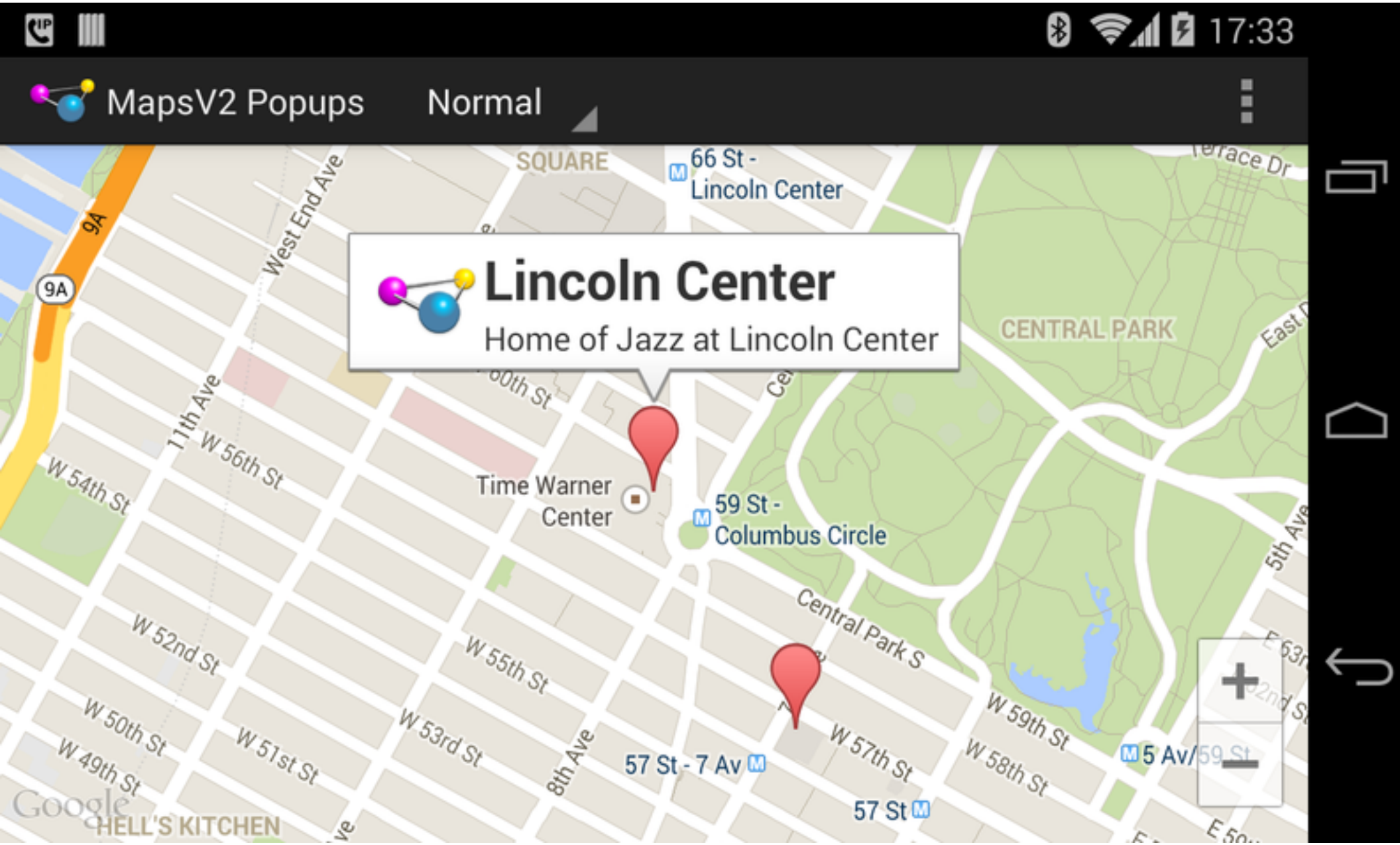

17:31
MapsV2 Flat Markers
Normal
CENTRAL PARK
Time Warner Center
59 St - Columbus Circle
57 St - 7 Av
57 St
5 Av/59 St
7 Av
50 St
50 St
W 65th St
W 63rd St
W 59th St
Central Park S
W 55th St
W 53rd St
W 52nd St
W 51st St
W 48th St
W 49th St
9th Ave
8th Ave
7th Ave
Broadway
9A
Google

Marking Up a Map

- Controlling the Info Windows
  - Implement InfoWindowAdapter interface
    - `getInfoContents()`: your own View to pour into Maps V2-supplied frame
    - `getInfoWindow()`: your own View with your own frame
  - Associate with GoogleMap via `setInfoWindowAdapter()`

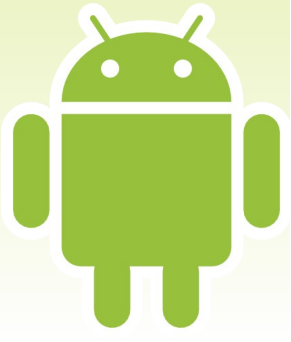
```
map.setInfoWindowAdapter(new PopupAdapter(getLayoutInflater()));
```

```
class PopupAdapter implements InfoWindowAdapter {
  LayoutInflater inflater=null;

  PopupAdapter(LayoutInflater inflater) {
    this.inflater=inflater;
  }

  @Override
  public View getInfoWindow(Marker marker) {
    return(null);
  }

  @Override
  public View getInfoContents(Marker marker) {
    View popup=inflater.inflate(R.layout.popup, null);

    TextView tv=(TextView)popup.findViewById(R.id.title);

    tv.setText(marker.getTitle());
    tv=(TextView)popup.findViewById(R.id.snippet);
    tv.setText(marker.getSnippet());

    return(popup);
  }
}
```


17:33
MapsV2 Popups
Normal
Lincoln Center
Home of Jazz at Lincoln Center
66 St - Lincoln Center
Time Warner Center
59 St - Columbus Circle
57 St - 7 Av
57 St
5 Av/59 St
CENTRAL PARK
HELL'S KITCHEN
West End Ave
11th Ave
8th Ave
5th Ave
Central Park S
W 54th St
W 56th St
W 52nd St
W 55th St
W 50th St
W 49th St
W 51st St
W 53rd St
W 57th St
W 58th St
W 59th St
9A
Google

Marking Up a Map

- Animating Marker Movement
  - Use ObjectAnimator to animate the position property of the Marker
  - Requires custom LatLngEvaluator to determine how to determine a LatLng that is X% of the way between start, end positions
    - Straight line interpolation
    - SphericalUtil.interpolate() for great circle calculations, plus dealing with longitude “wraparound”



```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
  if (item.getItemId() == R.id.animate) {
    animateMarker();

    return(true);
  }

  return(super.onOptionsItemSelected(item));
}
```

```
private Marker markerToAnimate=null;
private LatLng nextAnimationEnd=PENN_STATION;
```

```
private static final LatLng PENN_STATION=new LatLng(40.749972,
                                                    -73.992319);
private static final LatLng LINCOLN_CENTER=
    new LatLng(40.76866299974387, -73.98268461227417);
private static final LatLngBounds bounds=
    new LatLngBounds.Builder().include(LINCOLN_CENTER)
                              .include(PENN_STATION).build();
```

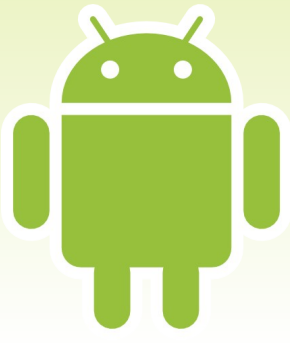
```
addMarker(map, 40.748963847316034, -73.96807193756104,
            R.string.un, R.string.united_nations);
markerToAnimate=
     addMarker(map, LINCOLN_CENTER.latitude,
                LINCOLN_CENTER.longitude, R.string.lincoln_center,
                R.string.lincoln_center_snippet);
addMarker(map, 40.765136435316755, -73.97989511489868,
            R.string.carnegie_hall, R.string.practice_x3);
addMarker(map, 40.70686417491799, -74.01572942733765,
            R.string.downtown_club, R.string.heisman_trophy);
```

```
private void animateMarker() {
  map.moveCamera(CameraUpdateFactory.newLatLngBounds(bounds, 48));

  Property<Marker, LatLng> property=
      Property.of(Marker.class, LatLng.class, "position");
  ObjectAnimator animator=
      ObjectAnimator.ofObject(markerToAnimate, property,
                              new LatLngEvaluator(), nextAnimationEnd);
  animator.setDuration(2000);
  animator.start();

  if (nextAnimationEnd == LINCOLN_CENTER) {
    nextAnimationEnd=PENN_STATION;
  }
  else {
    nextAnimationEnd=LINCOLN_CENTER;
  }
}
```

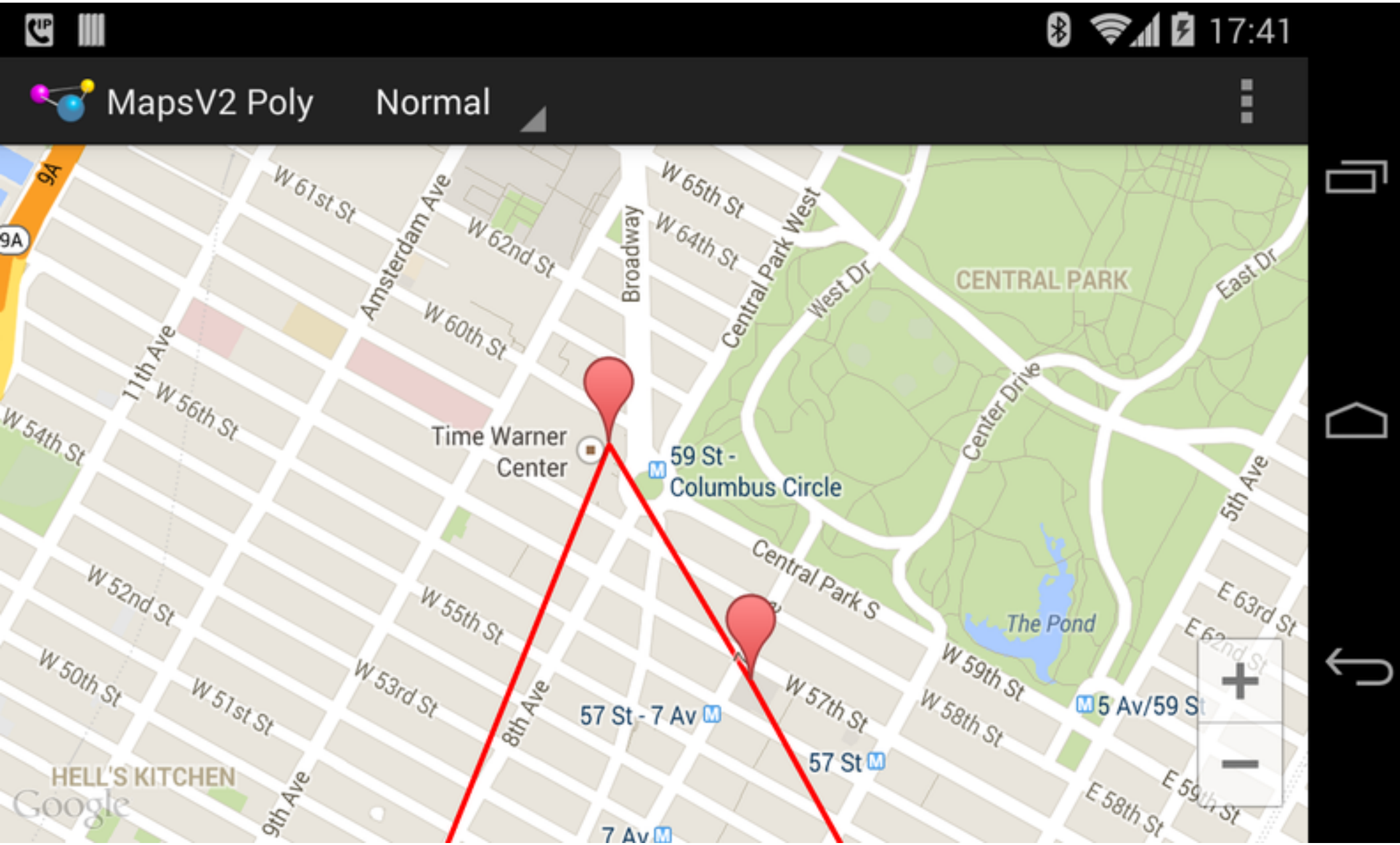
```
private static class LatLngEvaluator implements TypeEvaluator<LatLng> {
  @Override
  public LatLng evaluate(float fraction, LatLng startValue,
                         LatLng endValue) {
    return(SphericalUtil.interpolate(startValue, endValue, fraction));
  }
}
```

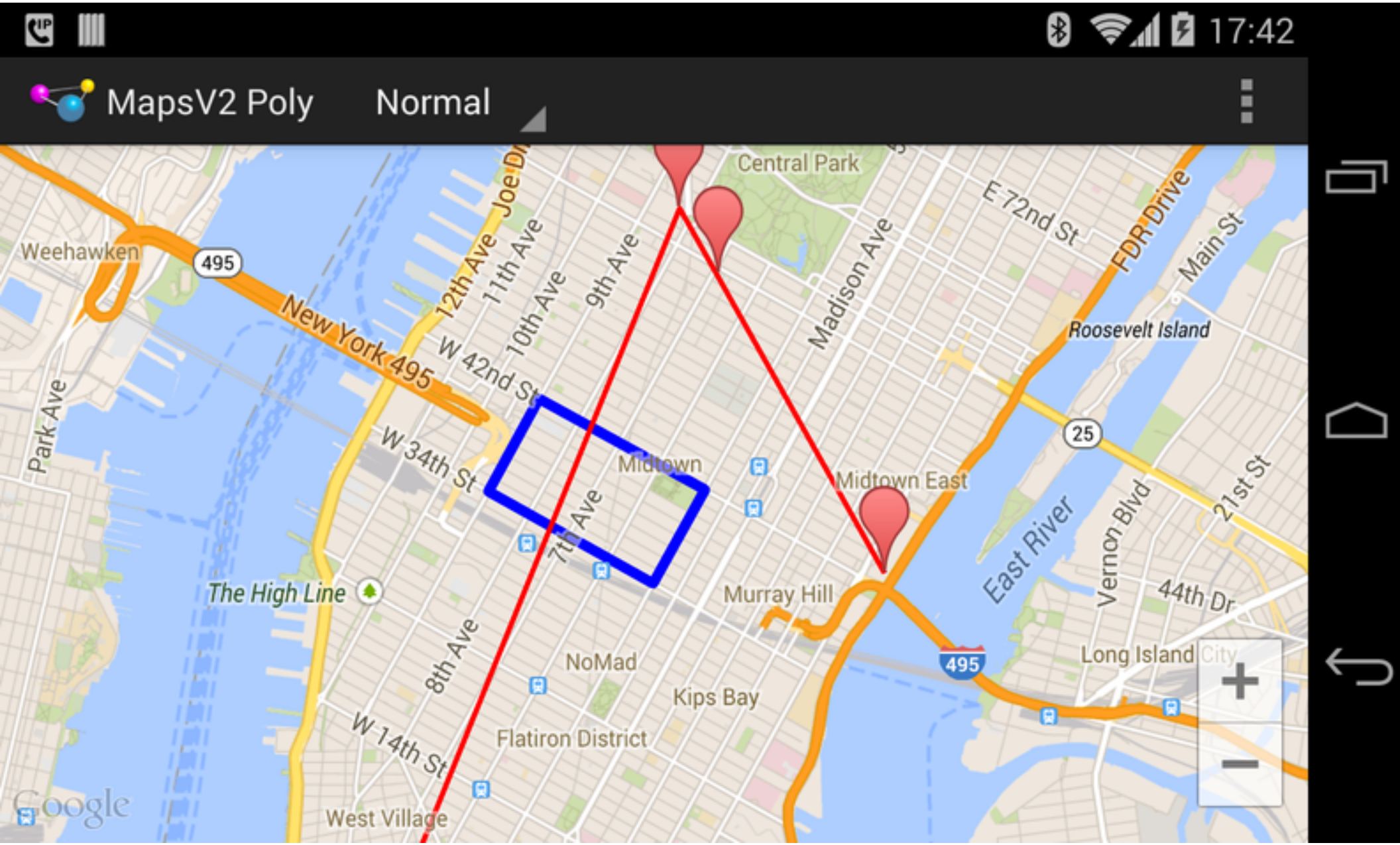
Marking Up a Map

- Polylines
  - You supply vertices, line details (color, thickness, etc.)
  - Map draws line between the vertices
- Polygon
  - You supply corners, fill details, map shades the area
- Circle
  - You supply center and radius, fill details
  - Map colors the circle



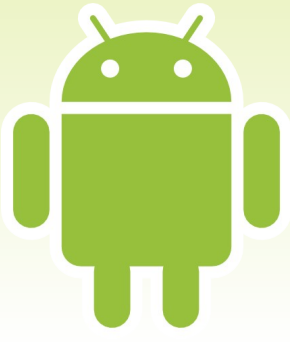
```
PolylineOptions line=
    new PolylineOptions().add(new LatLng(40.70686417491799,
                                         -74.01572942733765),
                              new LatLng(40.76866299974387,
                                         -73.98268461227417),
                              new LatLng(40.765136435316755,
                                         -73.97989511489868),
                              new LatLng(40.748963847316034,
                                         -73.96807193756104))
                             .width(5).color(Color.RED);

map.addPolyline(line);

PolygonOptions area=
    new PolygonOptions().add(new LatLng(40.748429, -73.984573),
                             new LatLng(40.753393, -73.996311),
                             new LatLng(40.758393, -73.992705),
                             new LatLng(40.753484, -73.980882))
                            .strokeColor(Color.BLUE);

map.addPolygon(area);
```


17:41
MapsV2 Poly
Normal
9A
W 61st St
Amsterdam Ave
W 62nd St
Broadway
W 65th St
W 64th St
Central Park West
West Dr
CENTRAL PARK
East Dr
W 60th St
11th Ave
W 56th St
W 54th St
Time Warner Center
59 St - Columbus Circle
Center Drive
5th Ave
Central Park S
The Pond
E 63rd St
W 52nd St
W 55th St
W 50th St
W 51st St
W 53rd St
8th Ave
57 St - 7 Av
W 57th St
W 59th St
W 58th St
5 Av/59 St
57 St
HELL'S KITCHEN
9th Ave
E 58th St
Google


17:42
MapsV2 Poly
Normal
Central Park
E 72nd St
FDR Drive
Main St
Roosevelt Island
Madison Ave
12th Ave
Joe DiMaggio
11th Ave
10th Ave
9th Ave
Weehawken
495
New York 495
W 42nd St
Park Ave
W 34th St
Midtown
7th Ave
Midtown East
25
21st St
East River
Vernon Blvd
44th Dr
The High Line
Murray Hill
Long Island City
8th Ave
NoMad
Kips Bay
W 14th St
Flatiron District
West Village
Google

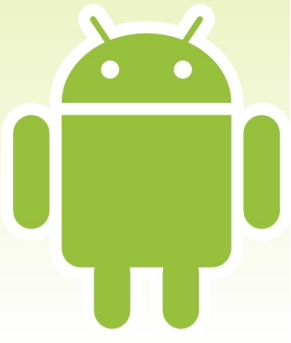
Maps and Locations

- Enabling “My Location”
  - `setMyLocationEnabled(true)` on GoogleMap
    - Adds “My Location” button
    - Requires suitable permissions (e.g., `ACCESS_FINE_LOCATION`)
    - When tapped, camera follows the user
- `setLocationSource()`
  - Feed in locations versus having Maps V2 use LocationClient itself
  - Useful if you want the location data too



Tracking Camera Changes

- setOnCameraChangeListener()
  - Call on GoogleMap
  - Pass in OnCameraChangeListener
  - Implement onCameraChange()
- Key Camera Attributes
  - Latitude and longitude
  - Zoom
  - Tilt



Pages of Maps

- MapFragment and ViewPager
  - It just works!
- Problem: ViewPager Wants Gestures
  - Default: cannot pan map horizontally
  - Solution: Custom ViewPager subclass, overriding `canScroll()`, to indicate widgets that handle their own scrolling

```
<com.commonsware.android.mapsv2.pager.MapAwarePager xmlns:android=
  android:id="@+id/pager"
  android:layout_width="match_parent"
  android:layout_height="match_parent">

  <android.support.v4.view.PagerTabStrip
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="top"/>

</com.commonsware.android.mapsv2.pager.MapAwarePager>
```

```
public class MainActivity extends AbstractMapActivity {
  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    if (readyToGo()) {
      setContentView(R.layout.activity_main);

      ViewPager pager=(ViewPager)findViewById(R.id.pager);

      pager.setAdapter(buildAdapter());
    }
  }

  private PagerAdapter buildAdapter() {
    return(new MapPageAdapter(this, getSupportFragmentManager()));
  }
}
```

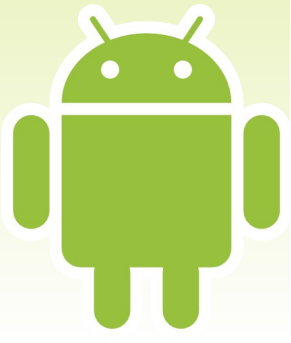
```
public class MapPageAdapter extends FragmentStatePagerAdapter {
  Context ctxt=null;

  public MapPageAdapter(Context ctxt, FragmentManager mgr) {
    super(mgr);
    this.ctxt=ctxt;
  }

  @Override
  public int getCount() {
    return(10);
  }

  @Override
  public Fragment getItem(int position) {
    return(new PageMapFragment());
  }

  @Override
  public String getPageTitle(int position) {
    return(ctxt.getString(R.string.map_page_title) + String.valueOf(position + 1));
  }
}
```

```
public class PageMapFragment extends SupportMapFragment {
  @Override
  public void onActivityCreated(Bundle savedInstanceState) {
    super.onActivityCreated(savedInstanceState);

    GoogleMap map=getMap();

    if (savedInstanceState == null) {
      CameraUpdate center=
          CameraUpdateFactory.newLatLng(new LatLng(40.76793169992044,
                                                   -73.98180484771729));
      CameraUpdate zoom=CameraUpdateFactory.zoomTo(15);

      map.moveCamera(center);
      map.animateCamera(zoom);
    }

    addMarker(map, 40.748963847316034, -73.96807193756104, R.string.un,
              R.string.united_nations);
    addMarker(map, 40.76866299974387, -73.98268461227417,
              R.string.lincoln_center, R.string.lincoln_center_snippet);
    addMarker(map, 40.765136435316755, -73.9798951489868,
              R.string.carnegie_hall, R.string.practice_x3);
    addMarker(map, 40.70686417491799, -74.01572942733765,
              R.string.downtown_club, R.string.heisman_trophy);
  }
```

```
public class MapAwarePager extends ViewPager {
  public MapAwarePager(Context context, AttributeSet attrs) {
    super(context, attrs);
  }

  @Override
  protected boolean canScroll(View v, boolean checkV, int dx, int x,
                              int y) {
    if (v instanceof SurfaceView || v instanceof PagerTabStrip) {
      return(true);
    }

    return(super.canScroll(v, checkV, dx, x, y));
  }
}
```

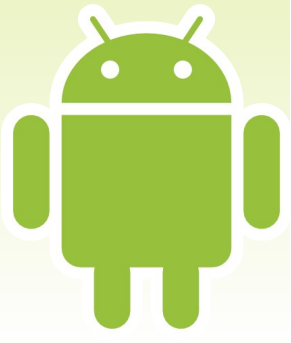
Helper Libraries

- android-maps-utils
  - From Google
  - Marker icons with text
  - SphericalUtil
  - Others



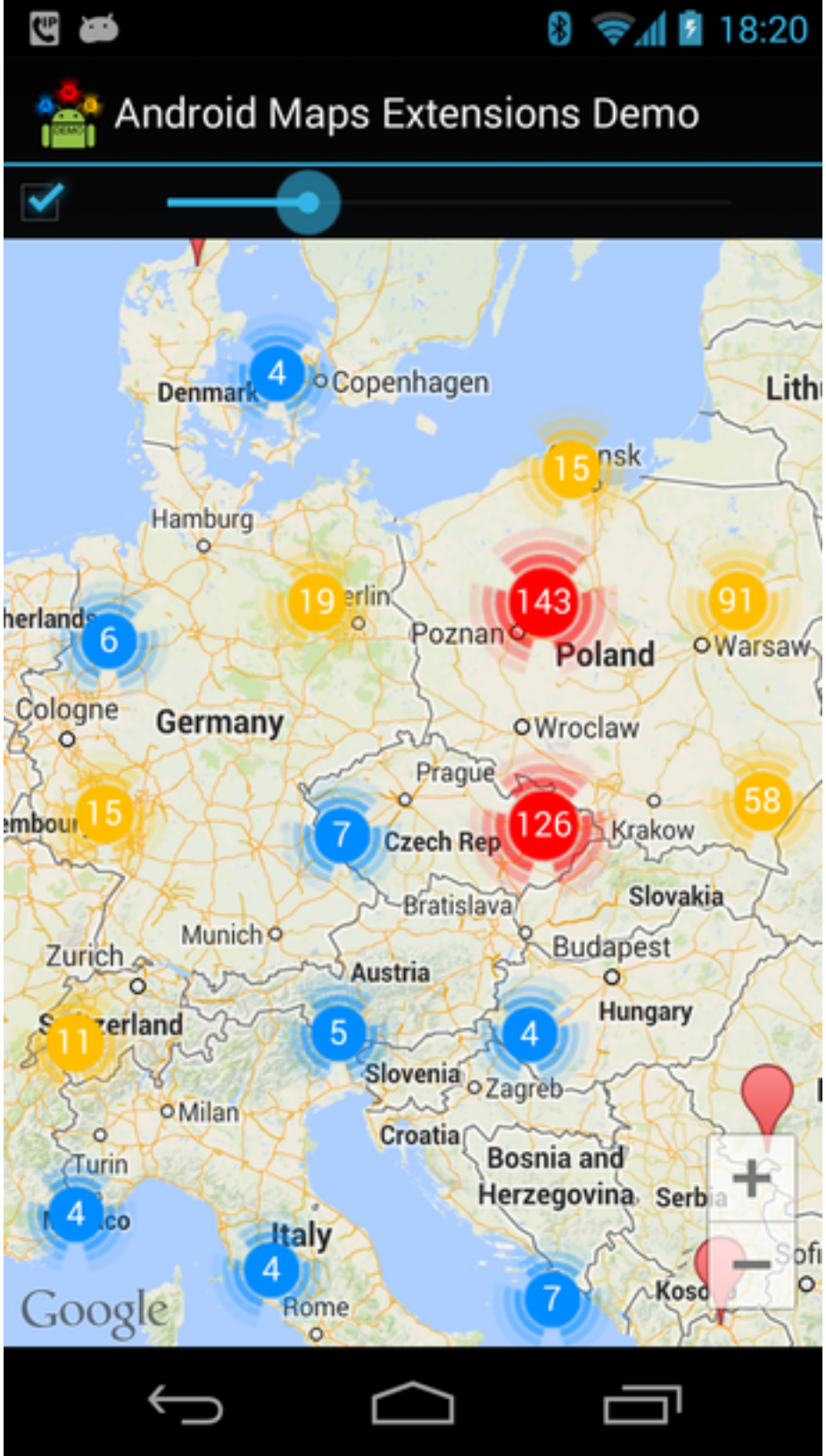
Helper Libraries

- Android Map Extensions
  - Marker clustering
  - Better model management



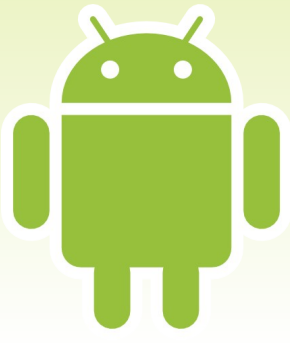

18:19
Android Maps Extensions Demo
Iceland
Sweden
Tunisia
Morocco
Algeria
Libya
Western Sahara
Mauritania
Mali
Niger
Chad
Nigeria
Guinea
Ghana
South Sudan
Google


18:20
Android Maps Extensions Demo
3
4
Iceland
Sweden
Finland
Norway
199
99
28
Ireland
Belarus
Poland
Germany
Ukraine
Austria
France
204
19
70
Spain
Greece
Turkey
Portugal
2
Tunisia
6
8
Algeria
Libya
Egypt
Western Sahara
Mauritania
Mali
Niger
Chad
Sudan
Burkina Fas
Guinea
Nigeria
Ghana
South Suda
4
Google


18:20
Android Maps Extensions Demo
4
Denmark
Copenhagen
15
Hamburg
19
143
91
6
Poznan
Poland
Warsaw
Cologne
Germany
Wroclaw
Prague
15
7
Czech Rep
126
Krakow
58
Slovakia
Bratislava
Munich
Budapest
Zurich
Austria
Hungary
11
5
4
Slovenia
Zagreb
Milan
Croatia
Turin
Bosnia and
Herzegovina
4
Italy
4
Rome
7
Google

Alternatives to Google Maps

- Nokia Maps
  - Used for Nokia X
  - Version of these used for Amazon Kindle Fire
- OpenStreetMap
  - “Wikipedia of maps”



Slides! And Other Stuff Too!

http://commonsware.com/webinars/advMapsV2.html