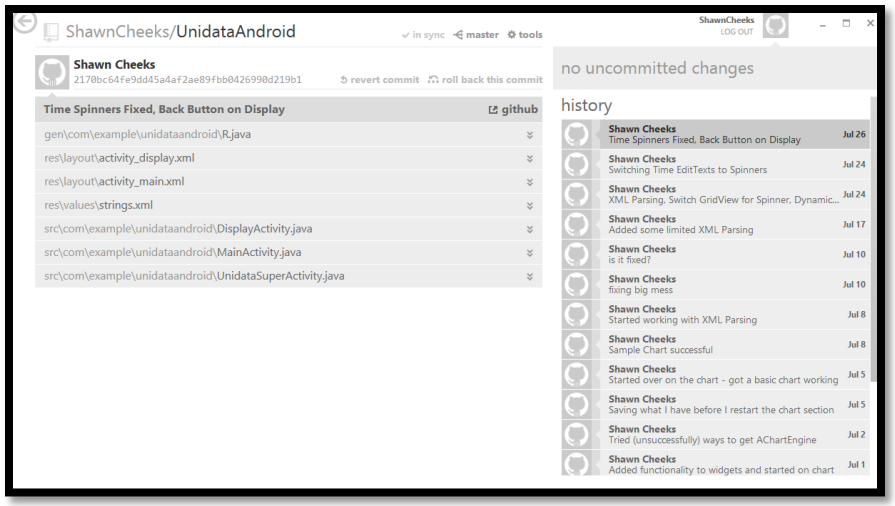
UNIDATA ANDROID

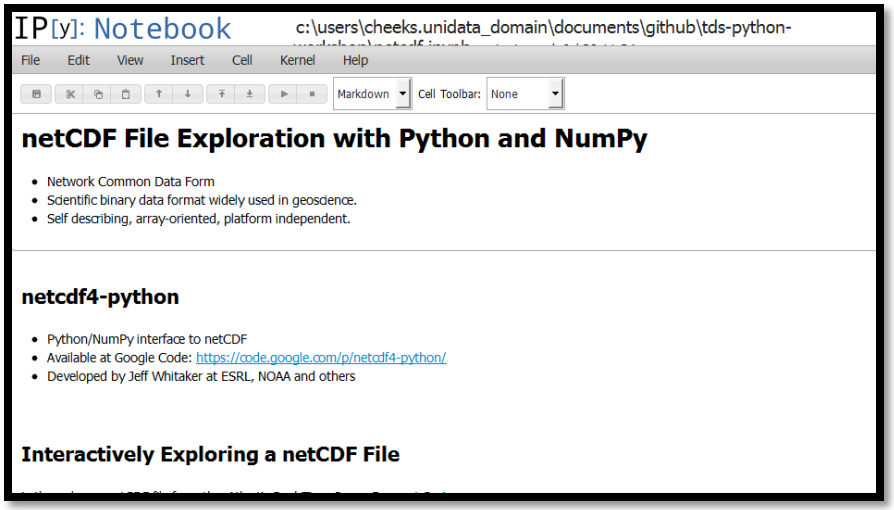
"What I Did This Summer"

Getting Acquainted



GitHub

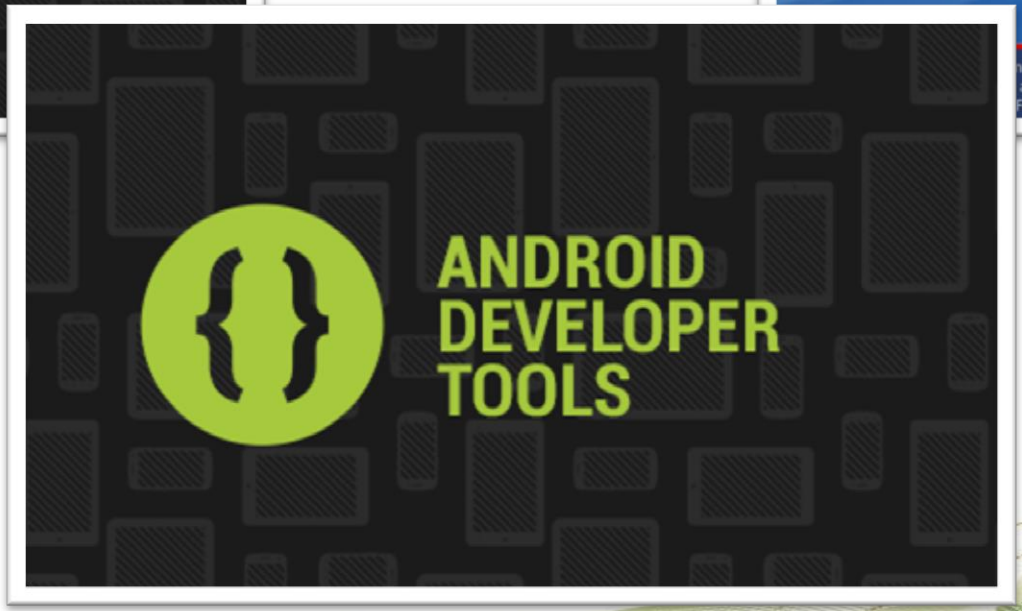
iPython Notebook



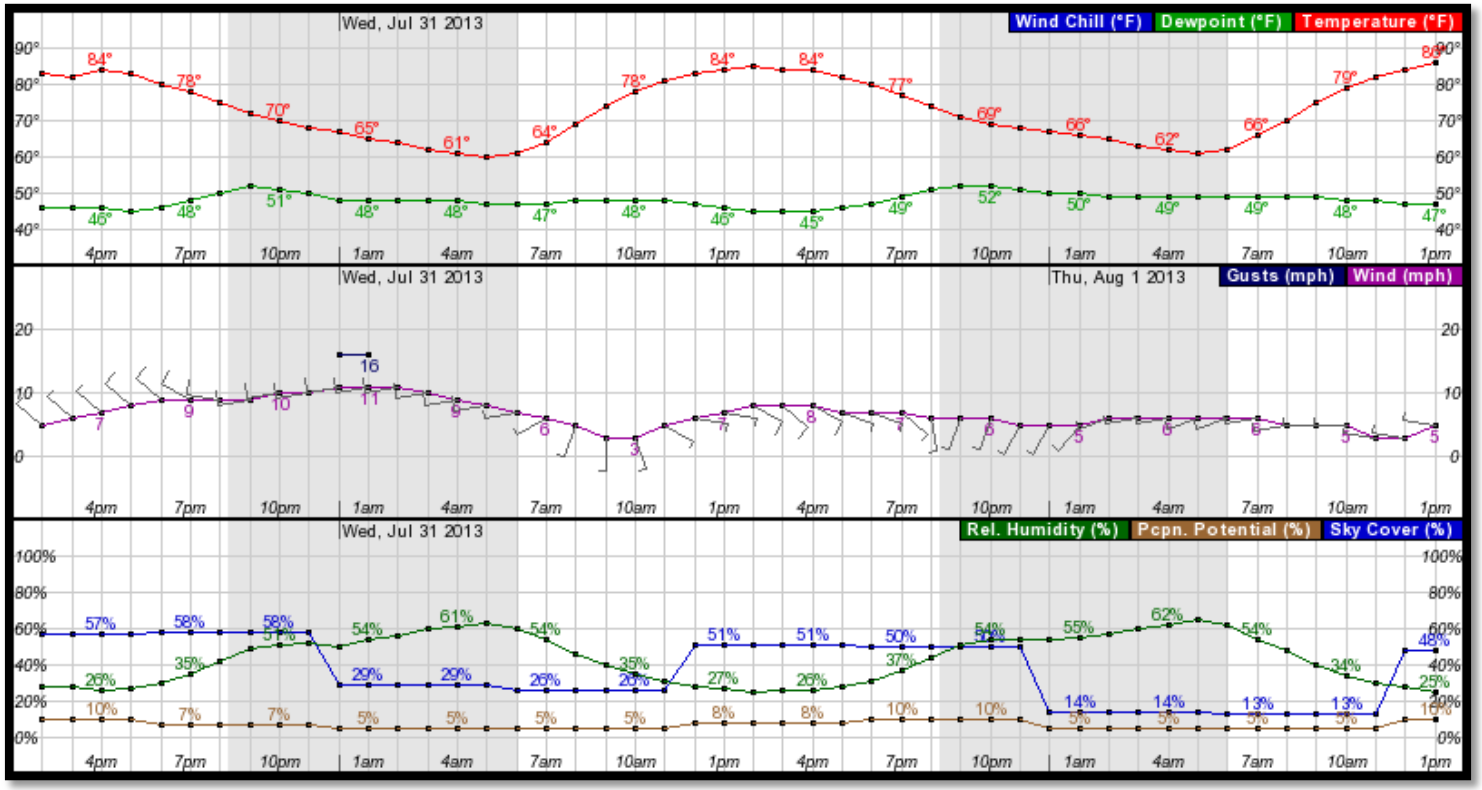
Finding a Project

- Met with Arnaud Dumont from RAL
- Weather Data in the Cockpit
- FAA Funded Research & Tablet App to Provide Weather Data to Pilots
- WebApp using PhoneGap

Settling on Android



Settling on Android



Time Series Forecast by NWS

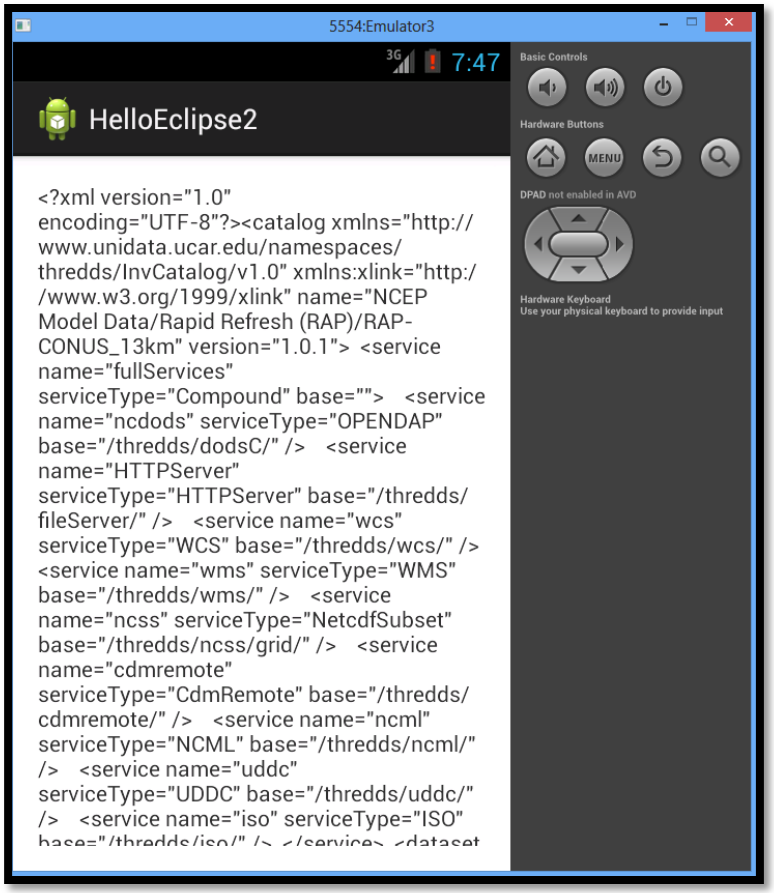
Attempts at netCDF

```
try {
    String remoteDataset =
            "thredds:resolve:http://thredds.ucar.edu/thredds/catalog/grib/NCEP/RAP/CONUS_13km/files/latest.xml";
    NetcdfFile ncd = NetcdfDataset.openFile(remoteDataset, null);
    List<Attribute> globalAttrs = ncd.getGlobalAttributes();

    String testMessage = "";
    for (Attribute attr : globalAttrs) {
        testMessage = testMessage + "\n" + attr;
    }
    ncd.close();
    hello.setText(testMessage);
} catch (IOException e) {
    e.printStackTrace();
}
```

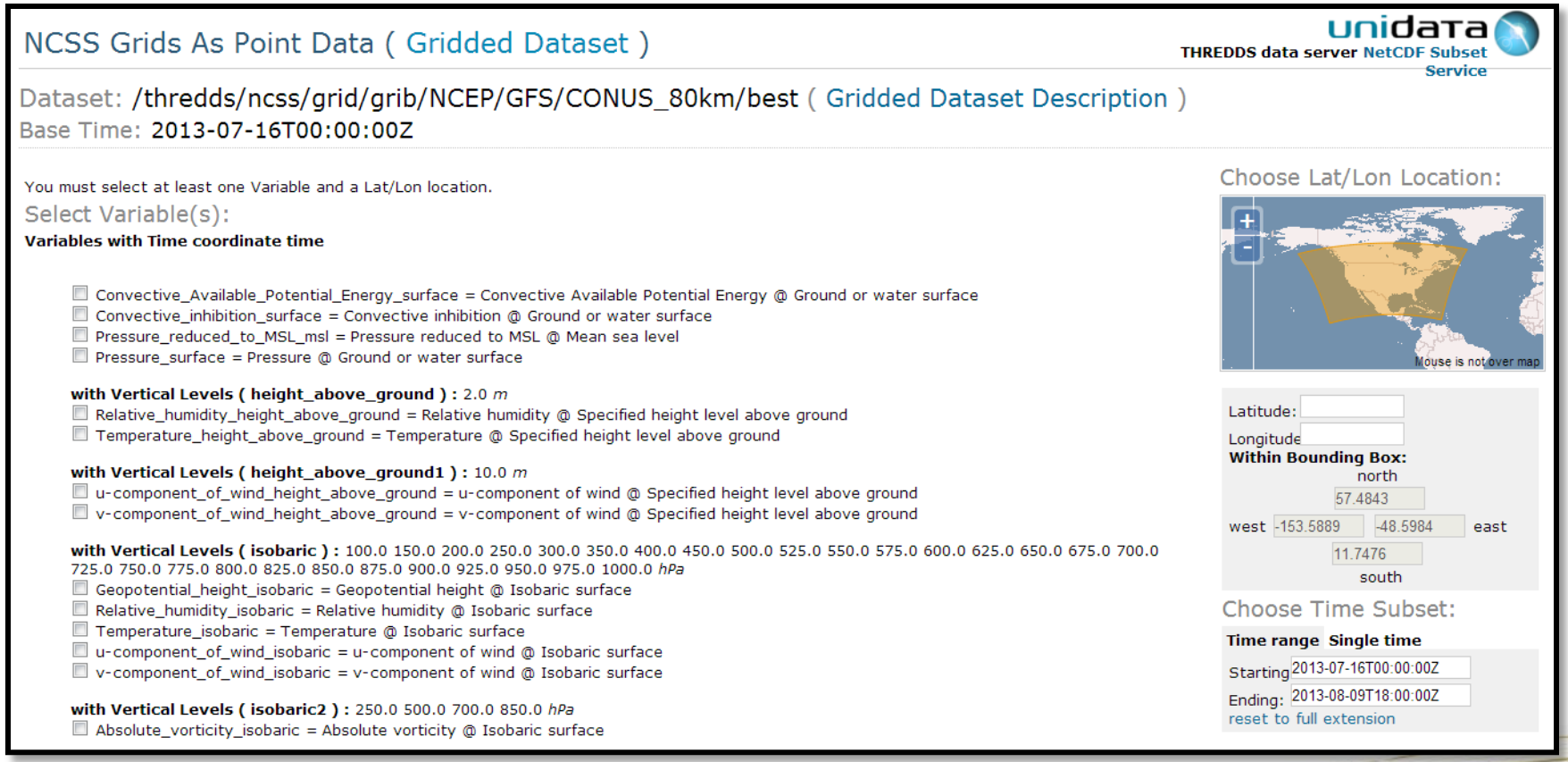
"Too many methods: 111248; max is 65536"

Formatting a URL



Example XML from URL in Emulator

Formatting a URL



NCSS Grids As Point Data (Gridded Dataset)

unidata
THREDDS data server NetCDF Subset Service

Dataset: /thredds/ncss/grid/grib/NCEP/GFS/CONUS_80km/best (Gridded Dataset Description)
Base Time: 2013-07-16T00:00:00Z

You must select at least one Variable and a Lat/Lon location.

Select Variable(s):

Variables with Time coordinate time

- Convective_Available_Potential_Energy_surface = Convective Available Potential Energy @ Ground or water surface
- Convective_inhibition_surface = Convective inhibition @ Ground or water surface
- Pressure_reduced_to_MSL_msl = Pressure reduced to MSL @ Mean sea level
- Pressure_surface = Pressure @ Ground or water surface

with Vertical Levels (height_above_ground) : 2.0 *m*

- Relative_humidity_height_above_ground = Relative humidity @ Specified height level above ground
- Temperature_height_above_ground = Temperature @ Specified height level above ground

with Vertical Levels (height_above_ground1) : 10.0 *m*

- u-component_of_wind_height_above_ground = u-component of wind @ Specified height level above ground
- v-component_of_wind_height_above_ground = v-component of wind @ Specified height level above ground

with Vertical Levels (isobaric) : 100.0 150.0 200.0 250.0 300.0 350.0 400.0 450.0 500.0 525.0 550.0 575.0 600.0 625.0 650.0 675.0 700.0 725.0 750.0 775.0 800.0 825.0 850.0 875.0 900.0 925.0 950.0 975.0 1000.0 *hPa*

- Geopotential_height_isobaric = Geopotential height @ Isobaric surface
- Relative_humidity_isobaric = Relative humidity @ Isobaric surface
- Temperature_isobaric = Temperature @ Isobaric surface
- u-component_of_wind_isobaric = u-component of wind @ Isobaric surface
- v-component_of_wind_isobaric = v-component of wind @ Isobaric surface

with Vertical Levels (isobaric2) : 250.0 500.0 700.0 850.0 *hPa*

- Absolute_vorticity_isobaric = Absolute vorticity @ Isobaric surface

Choose Lat/Lon Location:

Mouse is not over map

Latitude:
Longitude:
Within Bounding Box:
north 57.4843
west -153.5889 -48.5984 east
11.7476 south

Choose Time Subset:

Time range **Single time**

Starting 2013-07-16T00:00:00Z
Ending: 2013-08-09T18:00:00Z
reset to full extension

NetCDF Subset Service Request Form

App Components

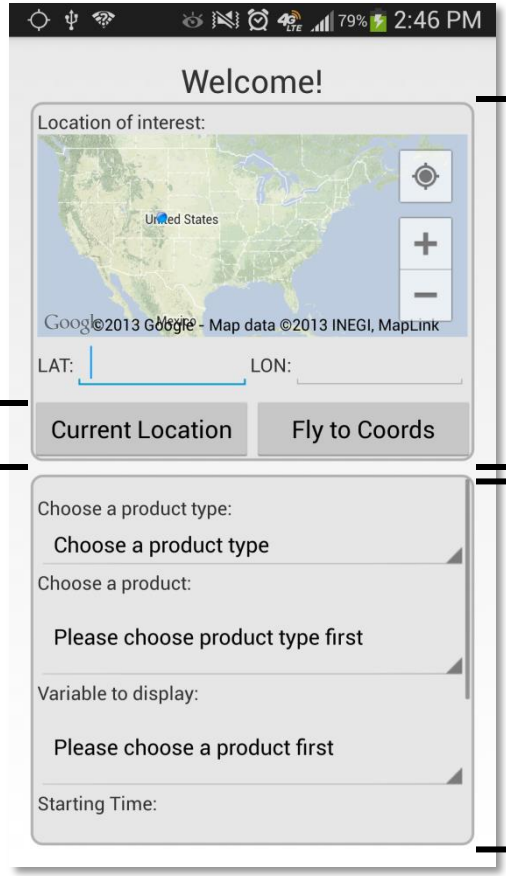
Data Request Needs	Satisfied By

• Product Type
  • Product
  • Variable(s)
	Spinners (hardcoded)

• Location
  • Lat
  • Lon
	Google Maps and Location Detection

• Time
  • Start
  • End
	Spinners (adaptive)

App Components

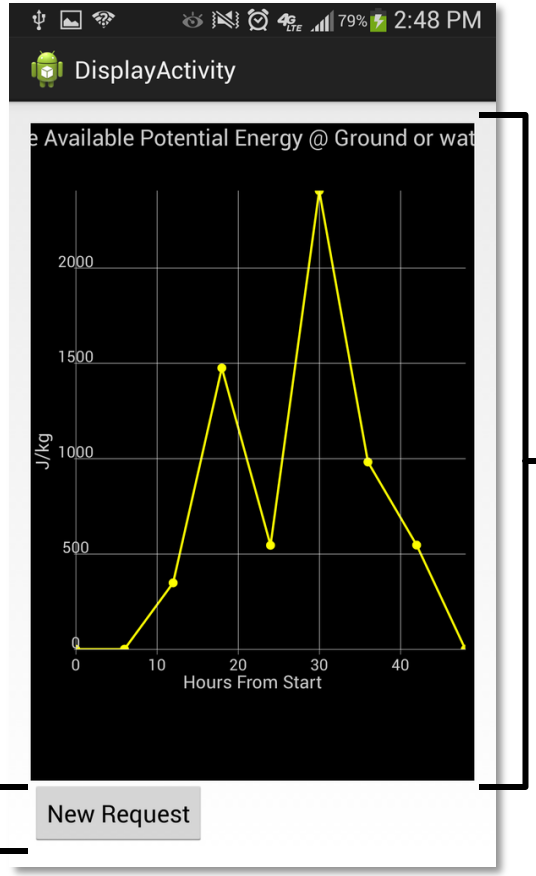


Location Services

Google Maps

Android Widgets

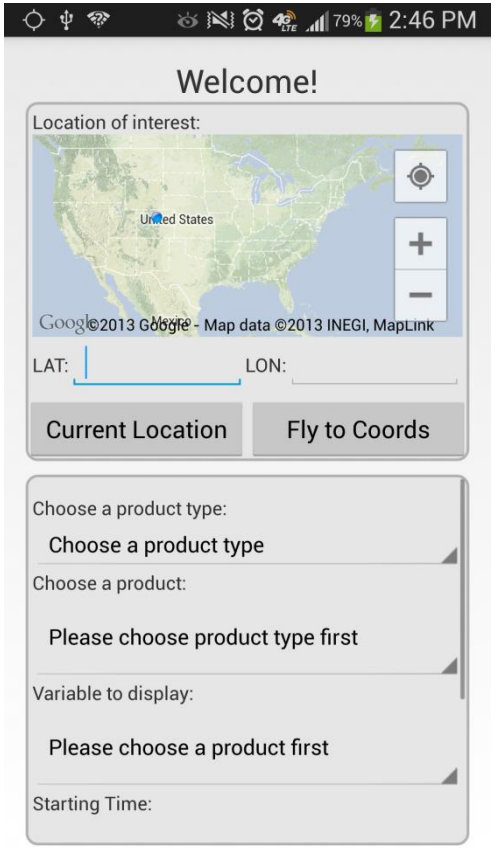
App Components



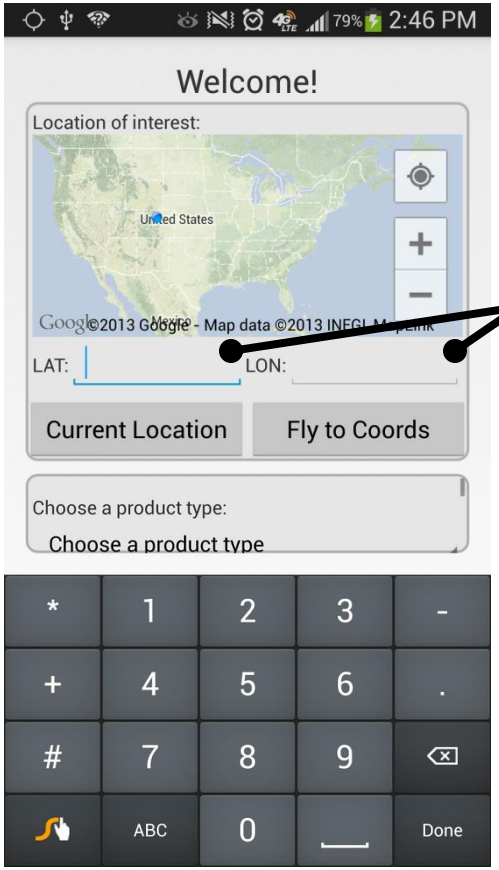
AChartEngine

Button

App Walkthrough

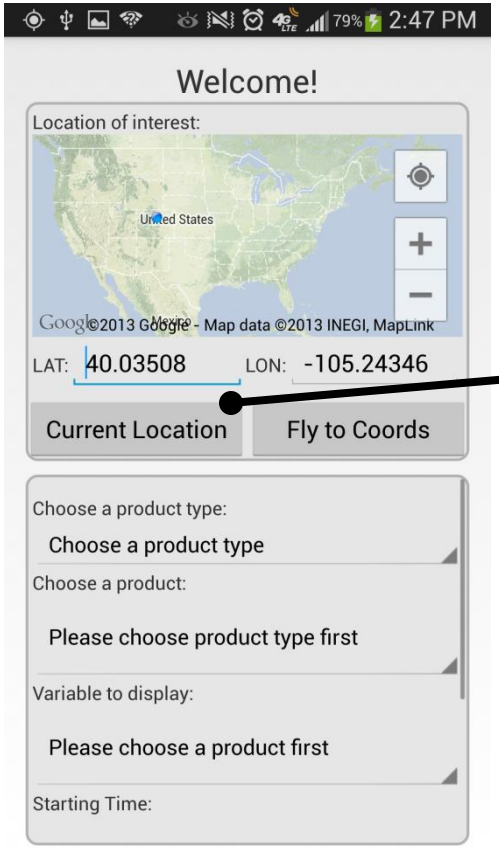


App Walkthrough



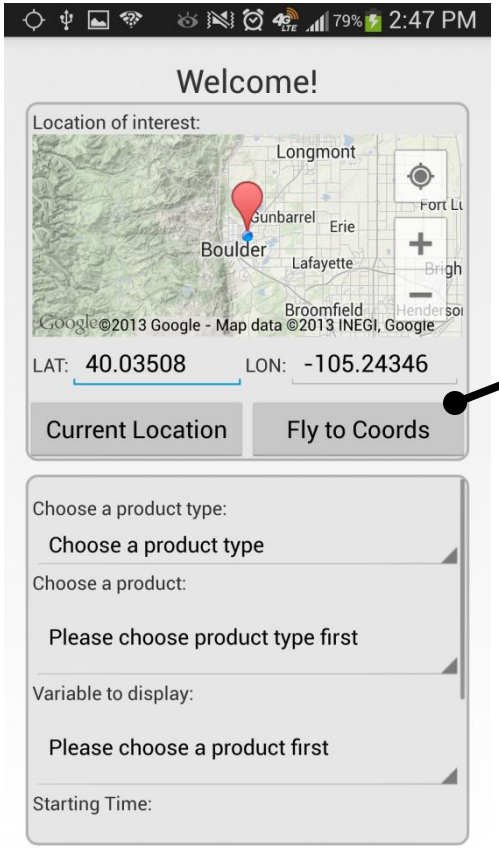
Lat & Lon are EditTexts, so the user may enter in coordinates manually

App Walkthrough



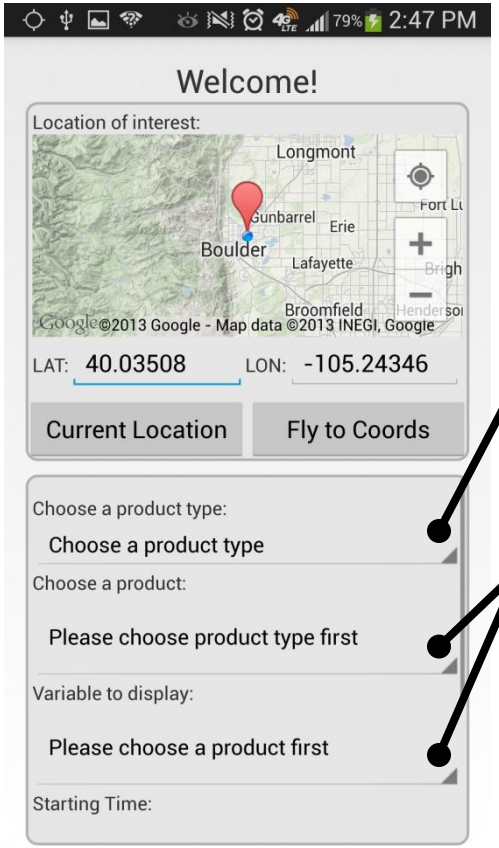
"Current Location" takes location and auto-fills the Lat & Lon variables

App Walkthrough



"Fly to Coords" zooms & centers on the entered Lat & Lon

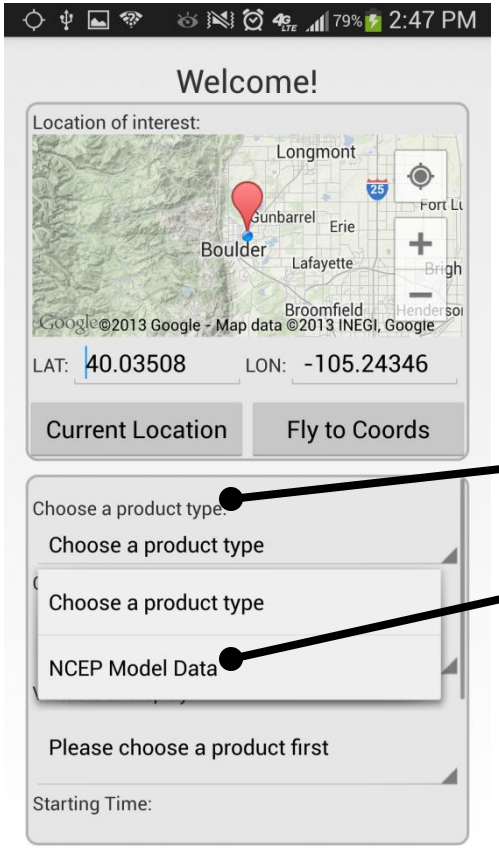
App Walkthrough



Only first spinner is unlocked

Other spinners require answers from earlier spinners

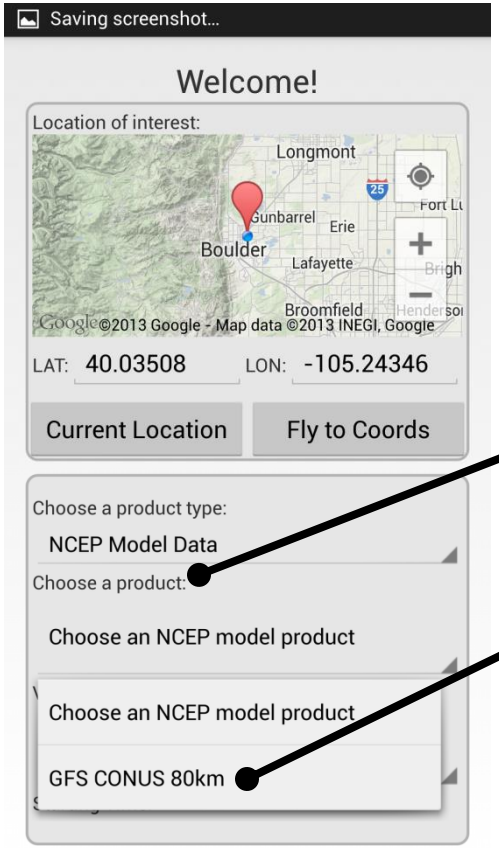
App Walkthrough



First is the “product type”

Currently hardcoded for only NCEP Model Data

App Walkthrough



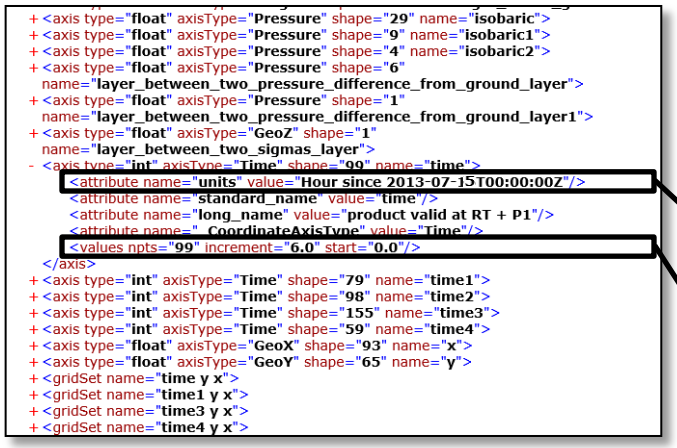
Once type is selected, the actual product is chosen

Currently hardcoded for GFS CONUS 80km model

App Walkthrough

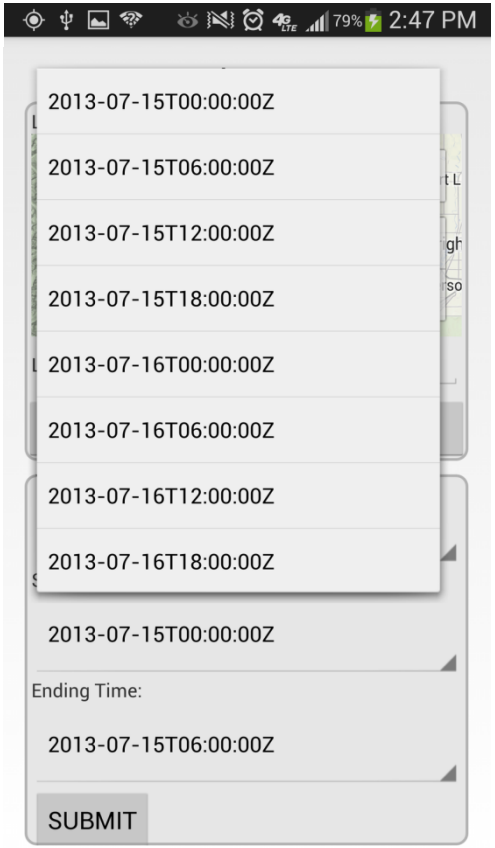
- Once product is selected, XMLPullParser takes over
- Parses the XML Doc for the product and returns the “time y x” variables
- The spinner is populated by the “time y x” variables

App Walkthrough



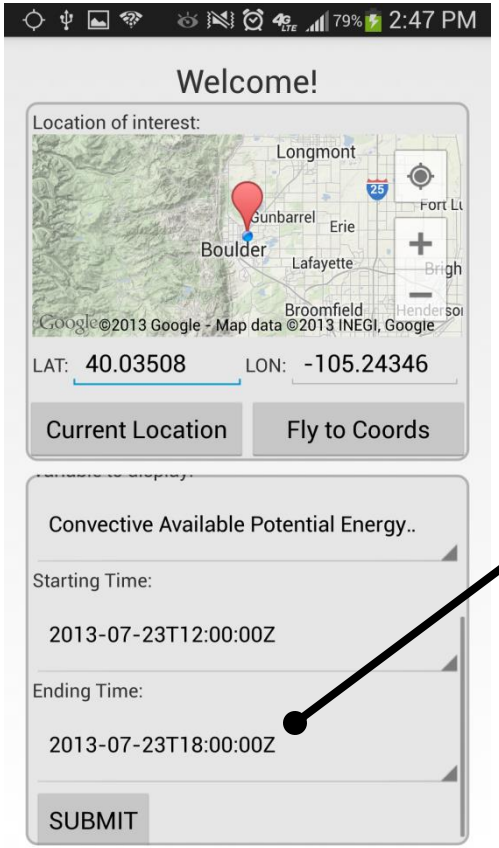
- The time options are also obtained from XML
- Starting time is defined by the time variable
- Time options are then incremented accordingly
- Ending time is at least one increment after selected start time

App Walkthrough



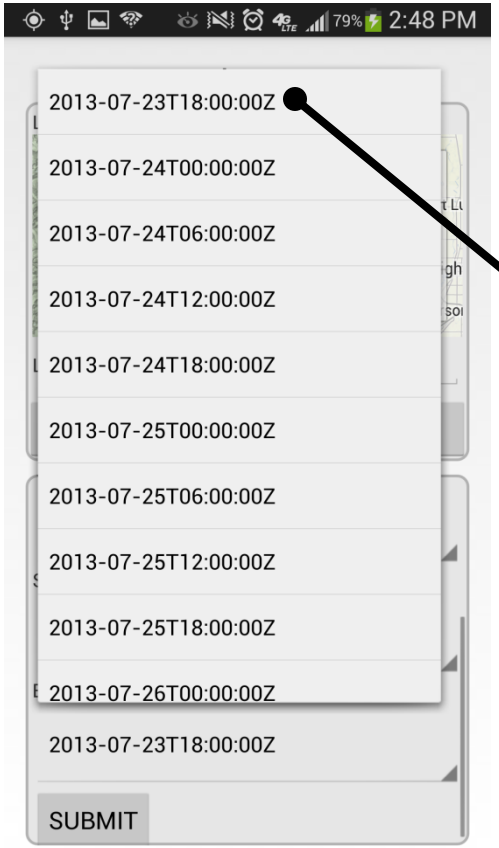
- Lots of time choices

App Walkthrough



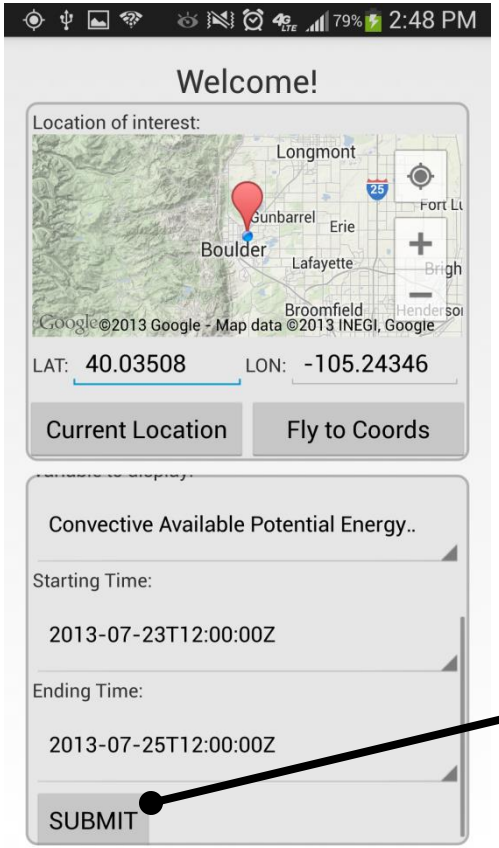
Again, the ending time automatically sets itself to be at least one frame after the starting time

App Walkthrough



The earliest end time choice is one increment after the selected start time

App Walkthrough



Now that all fields are completed…

Finding a Project

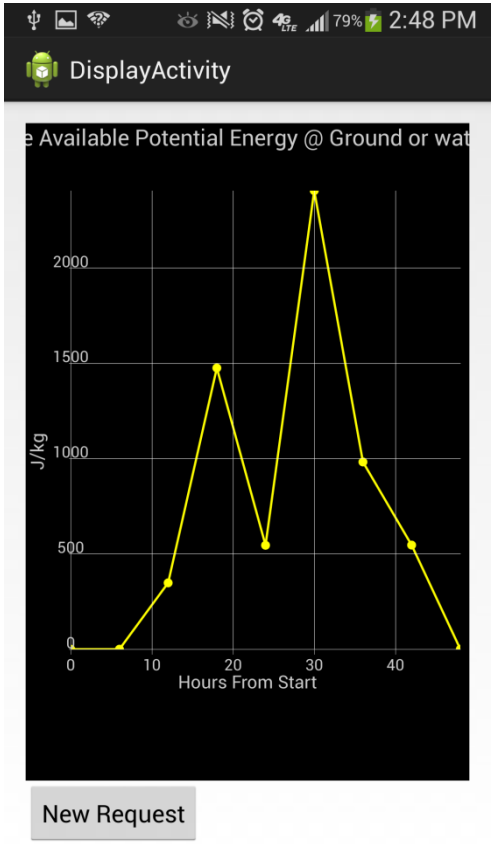
```
String a,b,c,d,e;
a=modelVariables[spinMainVariables.getSelectedItemPosition()].getName();
b=etMainLat.getText().toString();
c=etMainLon.getText().toString();
d=spinMainTimeStart.getSelectedItem().toString().replace(":", "%3A");
e=spinMainTimeEnd.getSelectedItem().toString().replace(":", "%3A");

super.setURL("http://thredds.ucar.edu/thredds/ncss/grid/grib/NCEP/GFS/CONUS_80km/best" +
             "?var=" + a +
             "&latitude=" + b +
             "&longitude=" + c +
             "&time_start=" + d +
             "&time_end=" + e +
             "&vertCoord=&accept=xml");
```

- The app creates a URL request to the server and calls the DisplayActivity

App Walkthrough

```
<?xml version="1.0" encoding="UTF-8"?>
- <grid dataset="/data/ldm/pub/native/grid/NCEP/GFS/CONUS_80km/GFS-CONUS_80km.ncx">
  - <point>
      <data name="date">2013-07-23T12:00:00Z</data>
      <data name="lat" units="degrees_north">40.03508</data>
      <data name="lon" units="degrees_east">-105.24346</data>
      <data name="Convective_Available_Potential_Energy_surface" units="J/kg">0.0</data>
    </point>
  - <point>
      <data name="date">2013-07-23T18:00:00Z</data>
      <data name="lat" units="degrees_north">40.03508</data>
      <data name="lon" units="degrees_east">-105.24346</data>
      <data name="Convective_Available_Potential_Energy_surface" units="J/kg">0.0</data>
    </point>
  - <point>
      <data name="date">2013-07-24T00:00:00Z</data>
      <data name="lat" units="degrees_north">40.03508</data>
      <data name="lon" units="degrees_east">-105.24346</data>
      <data name="Convective_Available_Potential_Energy_surface" units="J/kg">348.0</data>
    </point>
  - <point>
      <data name="date">2013-07-24T06:00:00Z</data>
      <data name="lat" units="degrees_north">40.03508</data>
      <data name="lon" units="degrees_east">-105.24346</data>
      <data name="Convective_Available_Potential_Energy_surface" units="J/kg">1475.0</data>
    </point>
  - <point>
      <data name="date">2013-07-24T12:00:00Z</data>
      <data name="lat" units="degrees_north">40.03508</data>
      <data name="lon" units="degrees_east">-105.24346</data>
      <data name="Convective_Available_Potential_Energy_surface" units="J/kg">545.0</data>
    </point>
  - <point>
      <data name="date">2013-07-24T18:00:00Z</data>
      <data name="lat" units="degrees_north">40.03508</data>
      <data name="lon" units="degrees_east">-105.24346</data>
      <data name="Convective_Available_Potential_Energy_surface" units="J/kg">2405.0</data>
    </point>
```



- The DisplayActivity uses the XMLPullParser to extract the data.
- The data is stored into a double array
- The times are converted into “hours since” and stored in an int array
- This is then graphed by AChartEngine

Known Bugs

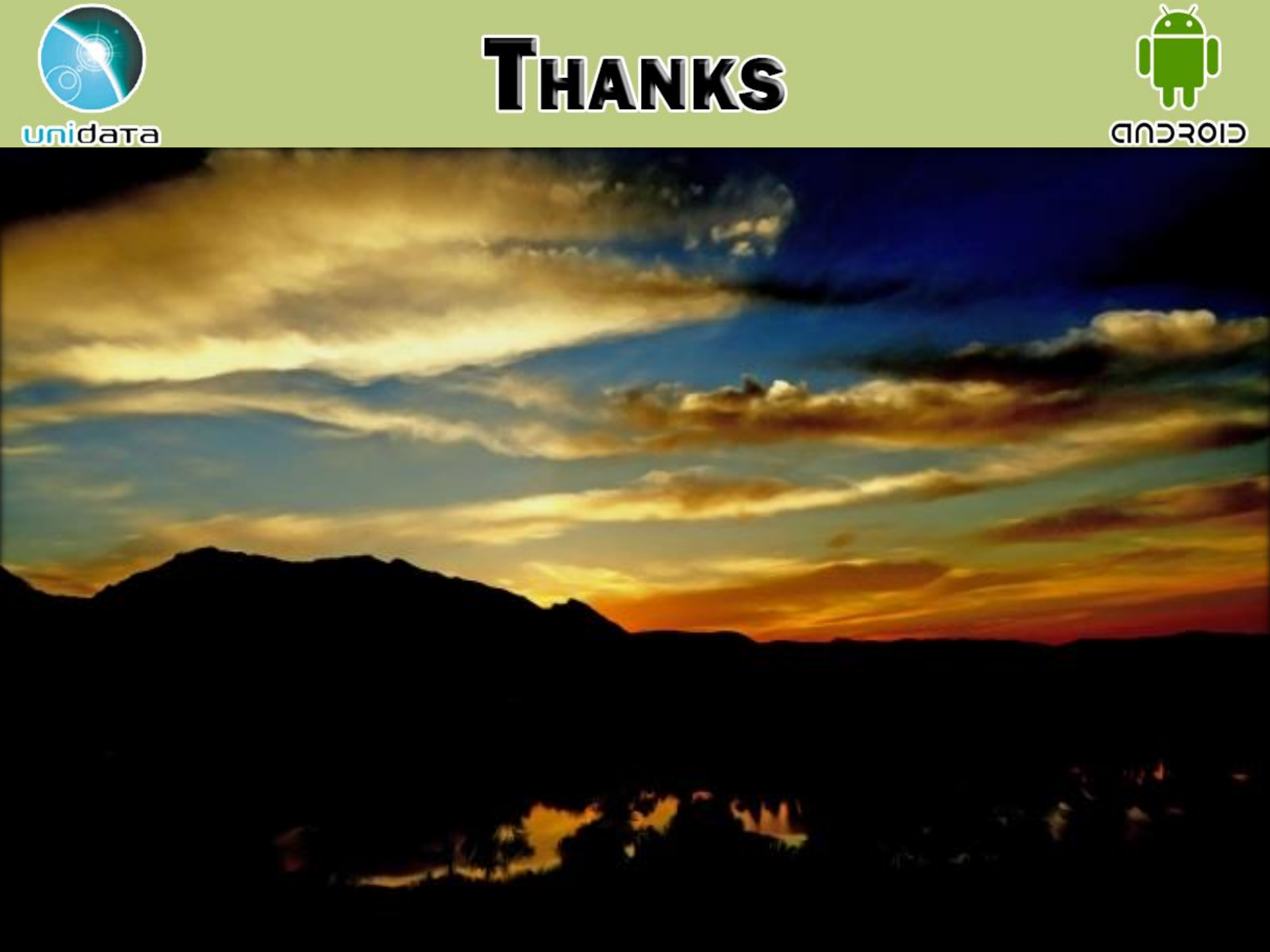
- Coordinate Boundaries not Enforced
- Will Accept Invalid URLs
- Landscape Orientation is Map-Only
- Native Units Only
- Long Variable Names

Next Two Days

- **Documenting Code**
- **Transferring Code**
- **Incorporating Last-Minute Suggestions**

And Beyond!

- More Variables, Models, & Products
- Multiple Graphs Simultaneously
- Fixing Known Bugs
- Finding New Bugs


Thanks
unidata
android