Pat Cain, KØPC / k0pc@arrl.net

APRS with a Smartphone

In an earlier article I presented a method for tracking QSO party mobile operators — or "rovers" — using online maps ("Generating Your Own QSO Party Mobile Maps" by Pat Cain, KØPC, Sep/Oct 2010 *NCJ*). This tracking relies on position reporting via the VHF automatic packet reporting system (APRS) network with connections to the Internet. Typically this has been done using the combination of a GPS, a terminal node controller, and a 2 meter FM transceiver. While this method works very well in some areas, it relies heavily on fixed stations to retransmit the position reports in order for them to reach a station that can connect to the Internet.

During my QSO party travels in Minnesota, Iowa and Wisconsin I ran across large rural areas that had no VHF APRS coverage at all. As a result, my tracking map had gaps of an hour or more between position reports in many areas. These gaps make the map much less useful for the operator who is attempting to track a rover from county to county.

Modern technology comes to the rescue. For the past couple of years I've owned a smartphone running the Android™ operating system. The phone contains all the hardware needed to report your position directly to the Internet as well as a GPS and a data connection. The only missing piece is an application to provide the smarts.

I searched the Android Market (now known as Google Play) for an application that can make APRS features available on my phone and settled on an application called *APRSdroid*, written by Georg Lukas, DO1GL. *APRSdroid* is a full-featured APRS application that uses the ARPS-IS network for position reporting (more on this below). *APRSdroid* provides the ability not only to report your position automatically but to track other nearby APRS-equipped stations. You can also exchange text messages with other APRS stations. I use *APRSdroid* simply to broadcast my position.

Program Setup

APRSdroid is simple to set up. Start up *APRSdroid* and select the MENU button on your phone, then choose the "More" option. Select "Preferences" to access the program's setup screen. You will need to enter your call sign and the service set identifier (SSID) you plan to use. The SSID is the number after your call sign. For example, I use an SSID of 9, so my APRS identification is KØPC-9. Most of the other preferences can be left at their default values to begin with.

APRS-IS Server

APRSdroid uses the APRS-IS server system to send and receive location data. The APRS-IS server is really a coordinated group of servers that share data. In order to connect to an APRS-IS server, you must first have a passcode issued by the program author. The first thing to do when you start up *APRSdroid* is to request a passcode for your call sign. If you have an APRS-IS passcode from another application, you can use it in *APRSdroid*. The Preferences screen contains an entry called "Request Passcode," which will bring up a Web form for submitting your information. The author may ask for verification that you hold an Amateur Radio license. It seems that using an e-mail of the form *<call sign>*@arrl.net makes the process easier and can speed the response. Once you receive your passcode, enter it in the Preferences screen.

The Preferences screen also has an entry for "Connection Preferences," where you can enter a server name. I use **rotate.aprs.net**, a round-robin DNS for all core APRS-IS servers. It will automatically direct you to one of the available servers.

Reporting Your Position

APRSdroid has several methods of reporting your position. You can generate a single-shot position report and leave it at that, but that's not useful when you're in motion. One automatic method involves sending a position report at a fixed interval that you define. This works but is not always terribly efficient. The best method is called SmartBeaconing™. This method is dynamic, based on your speed and direction. As you go faster the reporting interval is reduced, and if you stop moving the interval shifts to its maximum setting. In addition, SmartBeaconing also sends a position report when your heading changes. This makes sure you are tracked around corners.

Data Usage

This application uses your phone's data

Figure 1 — The *APRSdroid* "Hub" screen identifies stations and locations.

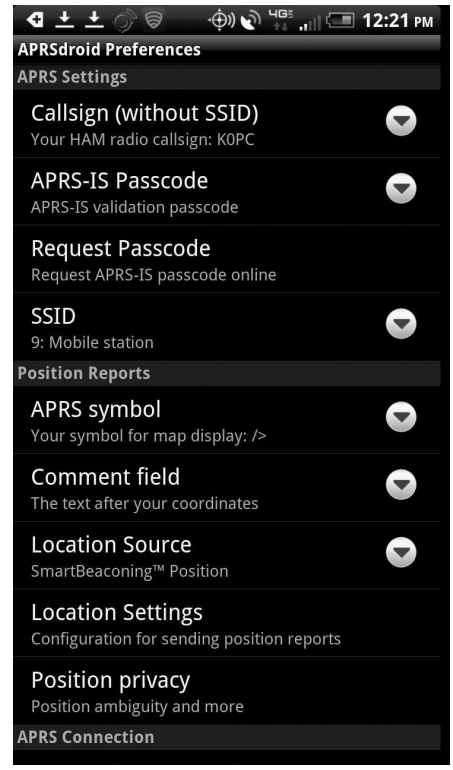

Figure 2 — A list of some available *APRSdroid* s "Preferences"

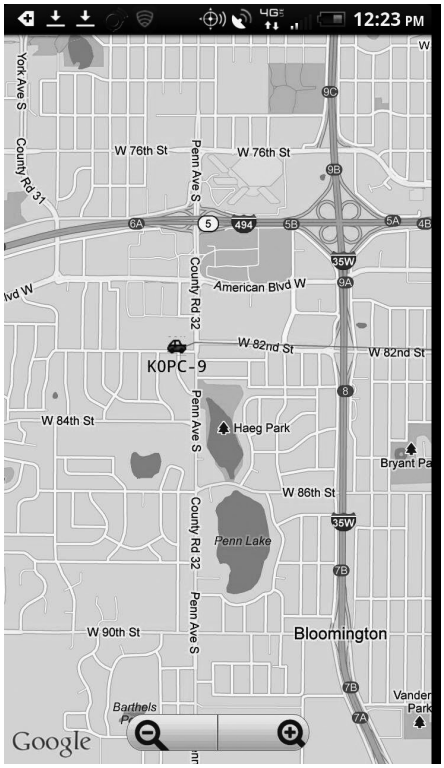

Figure 3 — A map showing the location of KØPC-9 as a tiny automobile graphic

network, so it is prudent to consider the data usage requirements. It would be a shame to have a great application, only to have it blow up your data plan. *APRSdroid* is very frugal on data usage. I have found it uses about 50 to 75 kB of data per hour. Over the course of a 12 hour QSO party this adds up to less than 1 MB of data.

One way to minimize the data the application uses is to reduce the "Neighbor Radius" setting on the "Connection Preferences menu." This setting determines the radius for nearby stations that the application is tracking.

Conclusion

APRSdroid greatly improved the usefulness of my QSO party tracking maps. Accurate location information lets the fixed operator know when the mobile station will cross into the next county. This program has worked very well for me. Georg is still enhancing it, and it is well worth more than its price, which is less than $5.

NCJ