APRS IP Mobile in a non-dynamic Wireless Environment

Or

802.11 APRS

Darryl SMITH, VK2TDS

Introduction

The APRS Internet Service is a wonderful concept for a highly distributed GIS system for connecting hardwired users to the worldwide collection of APRS gateways. However, the world is moving away from the idea of hardwired connections, and moving towards the situation where users are wishing to access the data of the iGate system whilst mobile.

While services such as FINDU.COM have the capability to send data to mobile devices such as laptops, PDA's and cell phones, this transmission of data is not necessarily bandwidth efficient – particularly when the maps are available already on the mobile device. Once colleague received a bill for about US$300 for his PDA for the month, with the majority from when he left FindU.COM running one night.

Whilst many of these devices allow a TCP/IP connection from the mobile device direct to the APRS Internet System (APRS-IS), the user is left with intermittent connections to the server – connections that are re-made when each new base station is encountered

Types of Networks

Two types of networks come to mind with the issue of changing IP addresses. The first is the GPRS network operated my the telco's, and the second is the 802.11 hotspots that are being installed world wide.

The GPRS networks based on the GSM telephone system are essentially classic TCP/IP using NAT and DHCP most commonly. When a user attaches to a new base station often a new NAT proxy will be used, and a new IP address will be assigned.

Alternately, consider the situation of a mobile 802.11 station driving through a city. The mobile will associate with a base station, and be assigned an IP address. The unit can then initiate a connection to the Internet until the signal drops out. After some period of time, another access point is found, and a new connection is initiated and the cycle repeats

Downloading data to the Mobile Unit

The Issue

In many networks, each separate base station or set of base stations are allocated some permanent IP addresses, and then use DHCP and NAT to assign private addresses to the user for use within range of that base station. When a user moves from one area to another, a new address is assigned, and a new connection to the APRS-IS must be established.

There are a number of consequences of this:

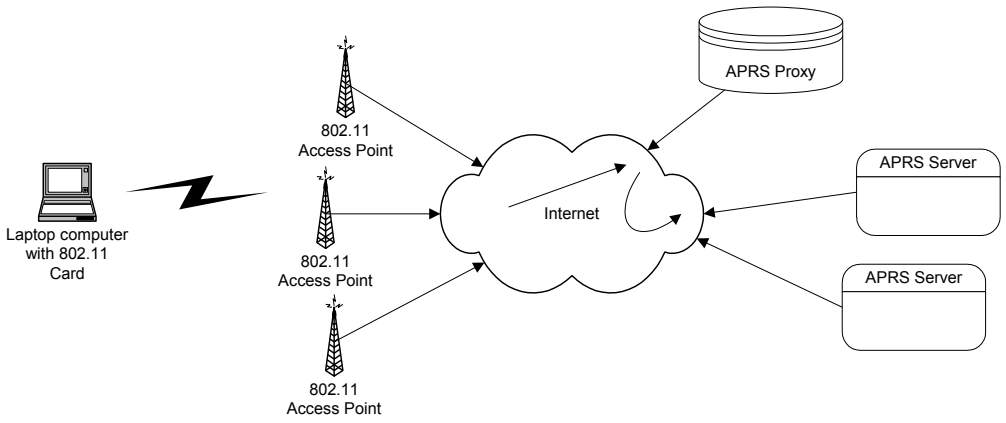
- Any packets during the transition are lost if no history dump is sent
- If a history dump is used, a large amount of data is sent, which may have already been sent.

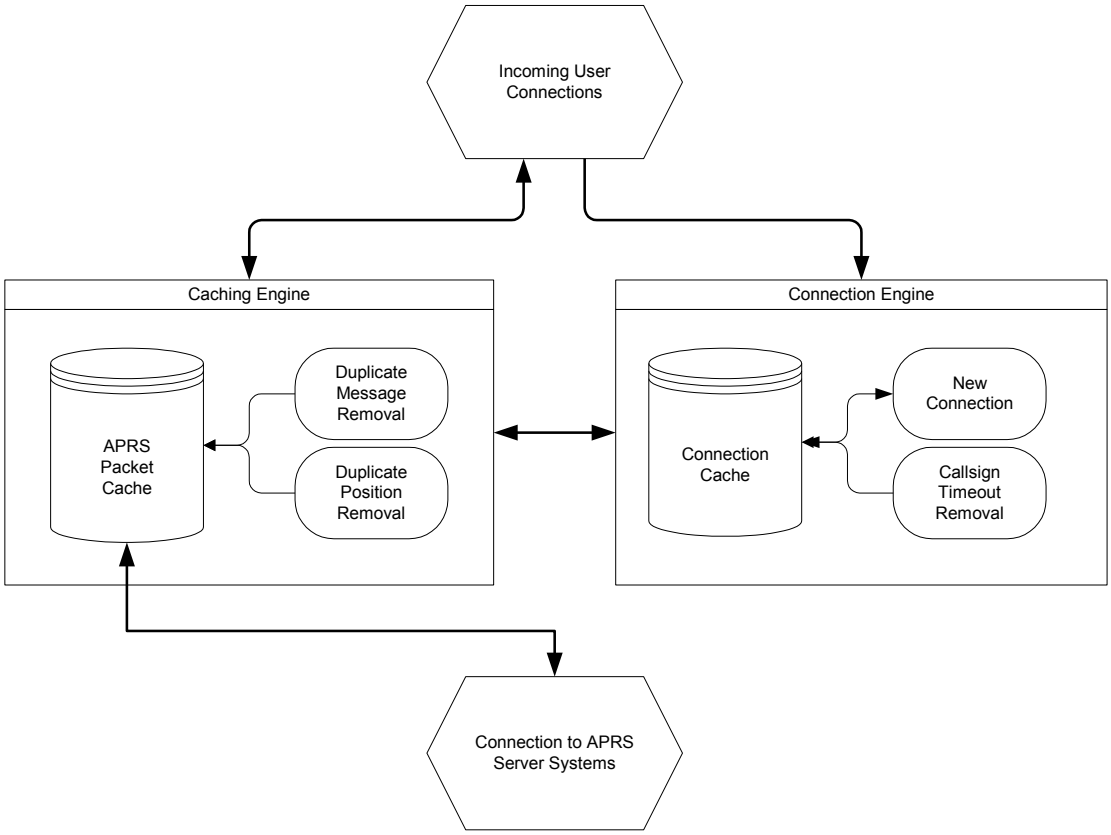
Neither situation is ideal

The Solution

In the ideal world, Mobile-IP would be used. This, however, relies on some coordination between base stations in order to operate. In our case this will not happen since all the base stations that we are using are out of our control. In the case of 802.11, some base stations may be under our control if operating under Part 97 rules.

The solution is to use an APRS caching proxy that acts as an interface between roaming clients and the APRS-IS; an interface that caches information between connections and sends a history report.





Technical Detail

1. Caching engine

The caching engine records all packets coming through and stores them into a FIFO buffer, each with a local timestamp. The size of the FIFO buffer will be a tradeoff between history length and memory utilization.

a. Duplicate Message Removal
When a new packet comes into the caching engine, a search is made to determine if other message packets can be removed from the cache. These would be duplicate message retries.

b. Duplicate Positions Removal
This is similar to the Duplicate Message Removal, but is for position reports. Implementation is optional since there are cases where it is useful for history purposes to be able to plot the positions of other stations over time.

2. Connection Engine

When a connection is made from a client, the callsign is stored in a list. Whenever a packet is sent to that client, a serial number is also stored next to that callsign.

a. Callsign Timeout

After a period of time, the caching engine will no longer hold all the packets that a particular user requires when they reconnect. When this happens the connection entry should be removed from the connection list.

b. New Connections

When a new connection comes in, a normal APRS login string will be sent containing the callsign of the user.

If the callsign is listed in the connection list, all the packets that are stored in the cache since the last connection are sent to the user. Once the TCP/IP status on the connection indicates that the data has transferred correctly, the serial number is to point to the top of the stack.

The New Connections function also closes any previous connection from the particular callsign. This is to solve the issue that is predominantly present with GPRS, but can also be present with 802.11 that connections may remain persistent when the mobile unit leaves the range of the base station.

Uploading Data

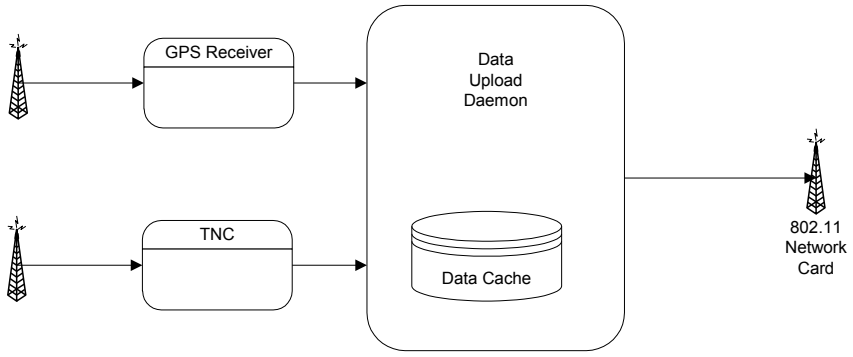
In most cases where existing software (such as UI-View) is used to upload data to the APRS-IS, beacons will be made whenever a TCP/IP connection is available. But this will only include the latest GPS position, and may not include other information that may need to be uploaded.

However in many cases there is a desire for tracking from a PDA without a requirement for displaying the positions of stations or for mapping. A small GPS daemon program could be written for a PDA or laptop to collect relevant data and upload it when network access is available.

The information that could be uploaded includes

- Real time position reports
- Position reports since the last successful upload to an access point. This is useful for where the intermediate positions are needed such as when an Radio Direction Finding (RDF) exercise is taking place
- Packets heard by a Mobile digipeater.

Since this is a dedicated application the software can be quite aggressive in re-establishing connections to upload data.



Uploading positions with HTTP

In private communications, author Robert Pepper noted that 'you should not discount the pervasiveness of HTTP access in the wireless environment'. He is quite correct. There are a growing number of wireless access points that only allow HTTP traffic. The solutions that I have proposed thus far will not work in this case.

Robert has proposed uploading data using HTTP GET requests for uploading and downloading the position of a single station. For instance

GET /yourpath&un=username&pw=password&cds=$GPRMC……
GET /yourpath&un=username&pw=password

This would return

GPSOK if in Remote mode
GPSOK$GPRMC…. If downloading a position
GPSERROR: if there is an error.

This system works well, although it is limited to tracking a single station. Without too many issues, it would be possible to extend the GET requests for uploading data from any number of stations by replacing the NMEA string with an APRS string, such as

GET /yourpath&un=username&pw=password&cds=VK2TDS>APRS:$GPRMC……

This still leaves the issue of getting position reports back. In that case it may

GET /yourpath&un=username&pw=password&md=mon

This would put the unit into MONITOR mode, and return position data whenever available by keeping the HTTP stream open

GPSOK:VK2TDS>APRS:$GPRMC….
GPSOK:VK2TKB>APRS:$GPRMC….
GPSOK:VK2TDS>APRS:$GPRMC….
GPSOK:VK2TDS>APRS:$GPRMC….

Intermediate Position Reports

One issue with this type of software is what should be done with intermediate position reports and other packets – information that would normally be discarded when more up to date information is presented. There are cases where the intermediate information may be required for historical purposes.

The recommendation is therefore to add the option for the user to be able to upload intermediate reports in chronological order as soon as a connection with the internet is re-established.

For packets that are time stamped, the issue of the currency of the data is self evident. Where no timestamps are included in the packet, I am recommending that a NMEA string with timestamp is uploaded at the same time as the encoded packets.

Conclusion

What we have seen is that there are some ways to allow the APRS system to be used in a highly mobile IP environment without resorting to Mobile IP protocols. Some of the caveats have been identified, and some areas for experimentation have been identified.