Public Service Event APRS Web Mapping for Wireless Devices

Ted Jacobson (W8KVK) and Rob Jacobson (KB9AFT)
for the 2011 TAPR and ARRL Digital Communications Conference
September, 2011

Abstract:

A case study describing real-time APRS web mapping support for Lead and Last runner tracking during the 2011 Athens Ohio Marathon. Objectives were: (a) to provide real-time web-based runner location maps for race officials, emergency services, and spectators without the use of JAVA or Flash; (b) to support a wide range of web-capable wireless devices; and (c) to optimize display speeds while reducing bandwidth demands.

Keywords:

APRS, internet, map, mapping, public service, wireless.

Introduction:

The Athens Ohio Marathon *[1]* is a 44 year old 26+ mile running event in the Appalachian hill country of Southeast Ohio. Much of the route is along a heavily forested, terrain obstructed rail-trail. Amateur Radio has supported this event for decades, with limited Automatic Position Reporting System (APRS) support introduced over the past few years. For the 2011 event, the APRS support team embarked on a service upgrade designed to provide real-time web-based Lead and Last runner location mapping for event officials, emergency services, and the general public. This experimental approach may also have value for other APRS field operations such as search and rescue, damage assessment, etc.
(Numbered footnote references *[ ]* are listed in Section 5.)

Contents Summary:

1. Designing and implementing the RF network
2. The APRS data to web mapping interface
3. The real-time web maps
4. Project evaluation and lessons learned
5. Numbered footnotes *[ ]*, Links, and References
6. Contacting the authors

1. Designing and implementing the RF network:

Significant terrain challenges made it desirable to do some preliminary Google Earth terrain obstruction signal contour "pseudo ray tracing" analyses *[2]* described by Bob Bruninga (WB4APR) as part of the annual Appalachian Trail Golden Packet Event *[3]*. These analyses provided useful approximations of where relay and fill-in digipeaters might be needed. Figure 1 shows an example of this type of analysis, clearly showing a major terrain obstruction approaching the KG6DI mobile digipeater location *[4]*.

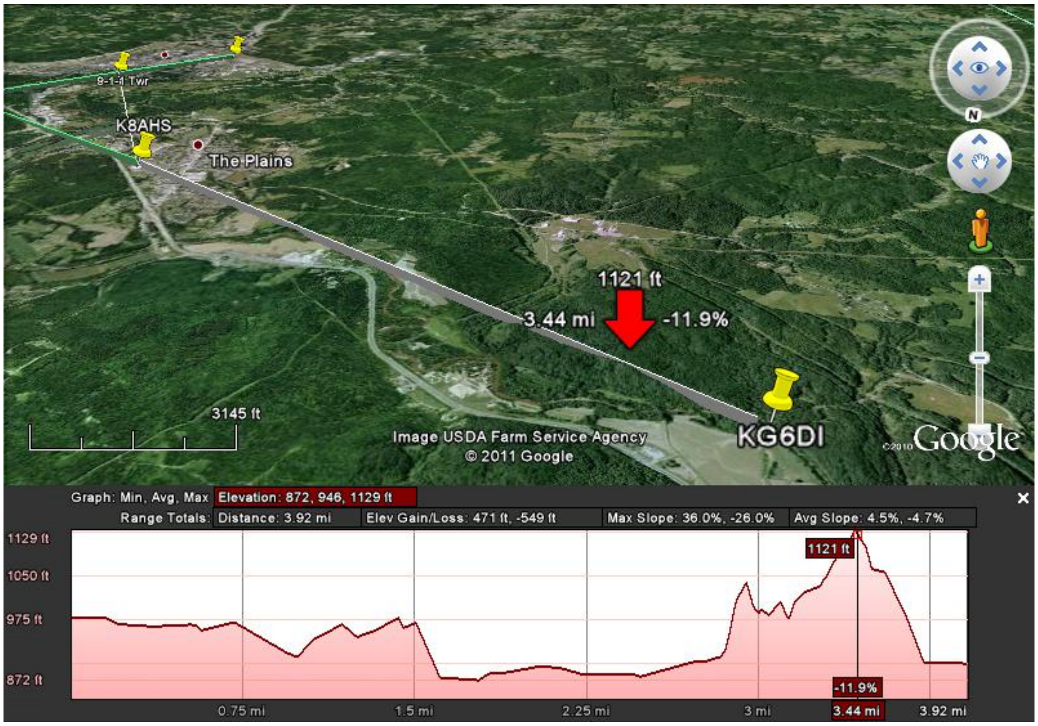

Figure 1 - Google Earth 'Ray Tracing' Analysis for the 2011 Athens OH Marathon route

The final network design incorporated the fixed digipeaters at W8UKE and K8AHS, with temporary mobile and portable digipeaters deployed elsewhere as needed. Figure 2 shows the final network.

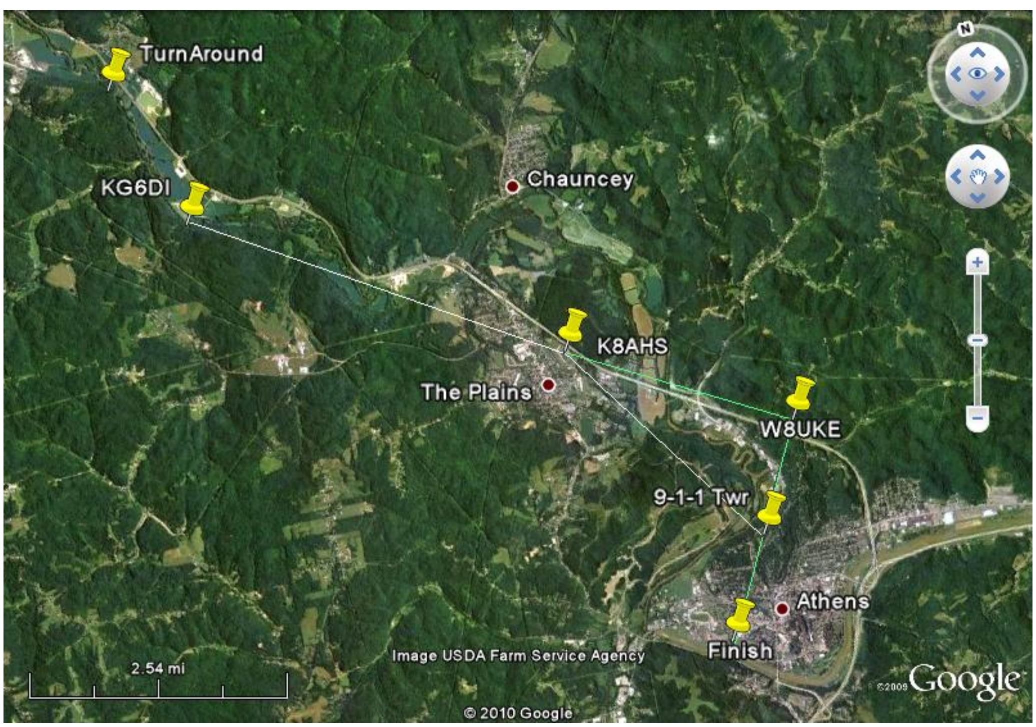

Figure 2 - Final APRS Network Architecture, 2011 Athens Ohio Marathon

The Lead and Last runner trackers were bicycle mobile stations using Byonics Micro-Trak "All-in-One" (AIO) units *[5]*. Pictures and narrative covering the Lead runner bicycle mobile station are available online *[6]* courtesy of WD8RIF. Final design of the RF network needed to take into consideration the low power and modest antennas of these trackers.

In addition to the two bicycle mobile stations, an APRS-equipped golfcart was also on the route for runner health and welfare coverage and supplies delivery. An over-the-air APRS display was provided at the finish line press box for use by event officials and the public address announcer.

The bicycle mobile stations transmitted beacons at thirty-five second intervals (:35) with different time-slotting values to prevent data collisions. The thirty-five second value was selected in order (a) to provide higher map plot resolutions, (b) to provide faster replacement of lost packets, and (c) to provide more precise position reporting for emergency services in the event of injury or other emergency. Much of this route is in deep forest not easily accessible from the public roads network.

The reason for the thirty-five second beacon interval (versus a thirty second interval) is that our local WIDEn-N configurations treat beacons generated at intervals thirty seconds or less as 'dupes', and thus discard such beacons.

WIDEn-N protocols *[7]* were used to insure that individual packets would reach the over-the-air APRS display at the finish line press box, regardless of which part of the network first received each packet.

2. The APRS data to web mapping interface:

The individual position reporting packets were received and ported to the internet via an internet-gateway (I-Gate) running UI-View32 software. The Application Programming Interface (API*) [8]* at APRS.fi *[9]* was chosen for the front-end data processing due to its robust performance and Extensible Markup Language (XML) generating capabilities. The APRS.fi API (properly) has strict limits on the number of queries it will accept from the same source within a specified time period. Abusing the API every thirty-five seconds for two bicycle mobiles (plus the golfcart) was avoided by (a) combining all three mobiles into a single XML query and (b) adding additional query repeat protection within the mapping interface itself.

3. The real-time web maps:

Due to the popularity and familiarity of Google Maps, a base map was created using Google Map API services. *[10]* This provided a "static" base map which could be presented to the end user from a local server, instead of repeatedly downloading the same map from Google. As with the APRS.fi API, Google has limits on how frequently each unique user may download the same map from its server.

Once created, this static base map was augmented with rail-trail Milepost (MP) numbers and time-stamp text boxes for runner icons. Large format runner icons and map references were employed to enhance viewing on small screens. The dark highlighting of the rail-trail itself was provided by Google in partnership with the Rails-to-Trails Conservancy *[11]*. Similar nationwide rail-trail and bicycle route map features are available on many Google Maps by selecting the 'bicycle' overlay.

The final online map as displayed to users is shown in Figure 3.

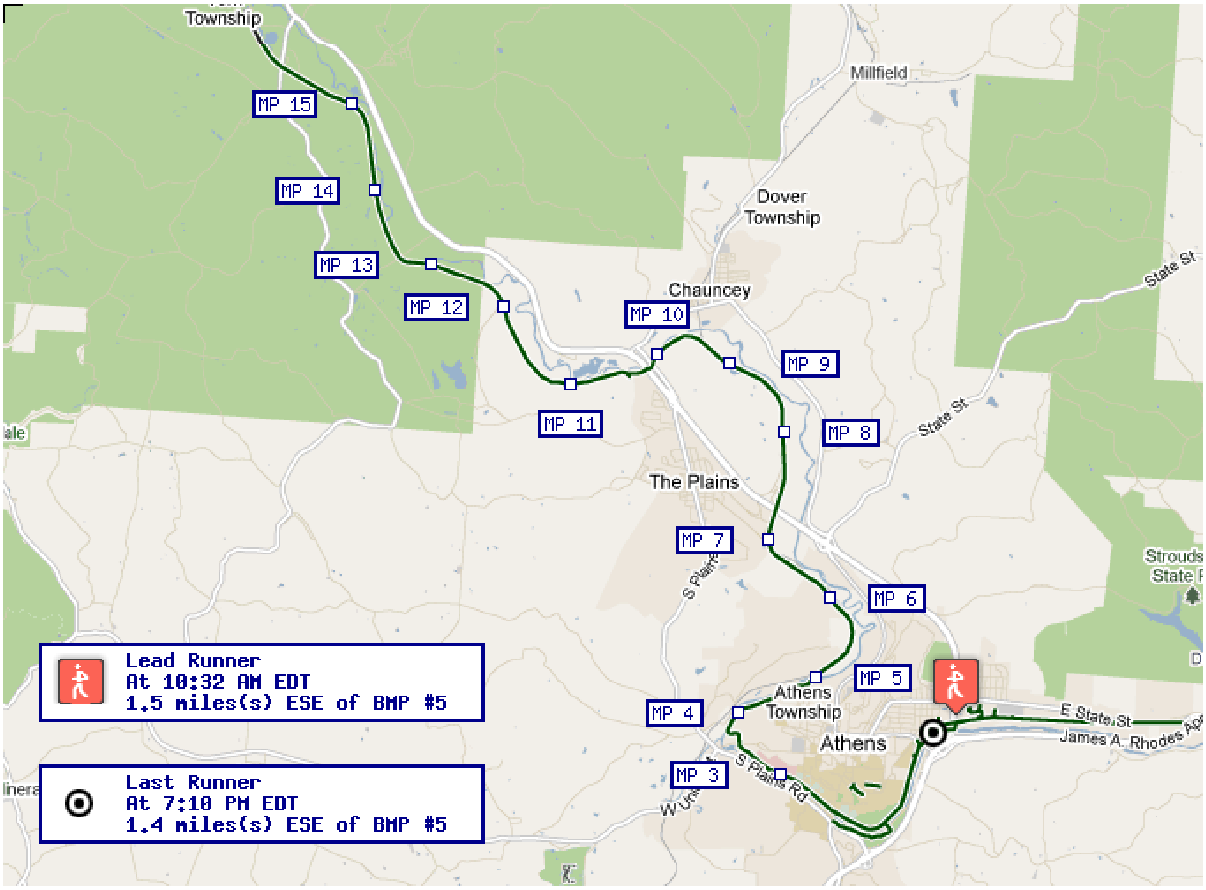

Figure 3 - Final online map as the end user received it

The wide range of web-enabled wireless device makes, models, vintages, and operating systems in use today, made it advisable to avoid JAVA or Flash in executing the final map display. The resulting final map, therefore, executed in pure HTML.

The actual APRS position data were extracted from the XML queries to the APRS.fi API, and then overlaid on the base map using PHP (a popular open source web scripting language). In the process, the icon placements were updated, as were the text references to distance to the nearest rail-trail milepost (MP). Since these steps are all 'server side' services, the end user's device did not have to perform any of the processing, thus significantly increasing map download speeds while reducing bandwidth demands. As more and more cellular carriers discontinue unlimited data plans, this bandwidth conservation feature will become more and more important.

For many (but not all) web-enabled wireless devices, the map was 'refreshed' via HTML every sixty seconds. The timestamps for each bicycle mobile were also updated if/as new data became available.

It should be noted that other types of (non-Google) maps could be used in a project of this kind, but any such map would have to be 'geo-referenceable', i.e. one must be able to determine the exact latitude/longitude value for each pixel on the map.

4. Project evaluation and lessons learned:

The RF network seemed to perform well. It may have been over engineered, but it was helpful to have some redundancy and coverage overlap in case of one or more site failures. A plot of all valid runner location beacons actually processed, appears in Figure 4, courtesy of APRS.fi.
Top to bottom distance of this route is 13+ miles.

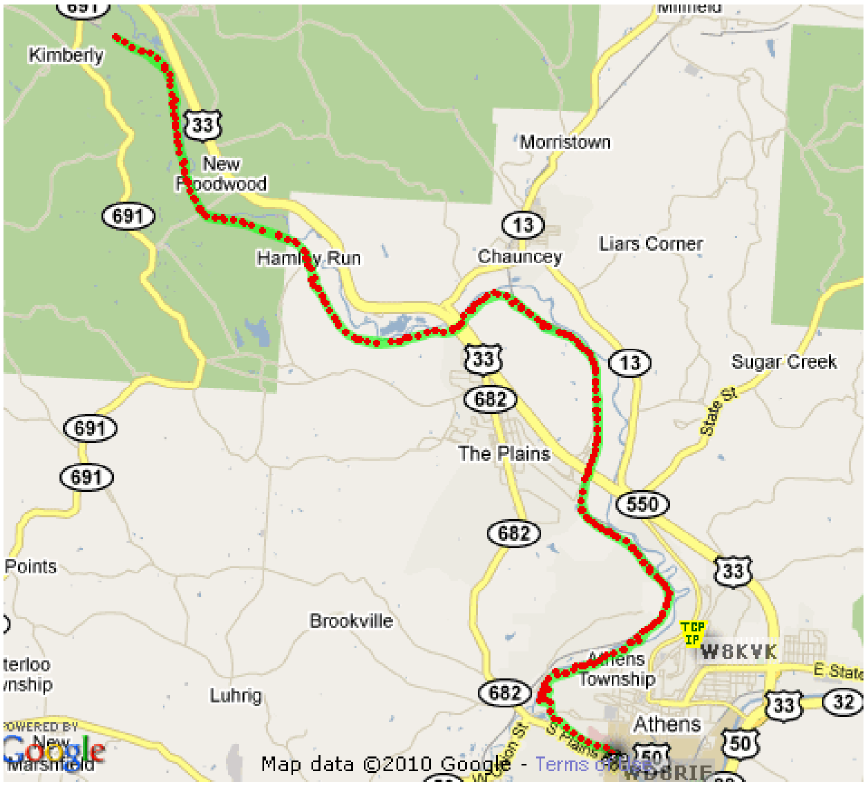

Figure 4 - Actual valid packets received and plotted (Courtesy APRS.fi)

The web mapping applications and user interfaces also performed well, although being our first year of publicized service, the number of simultaneous web users over the six hour event was moderate. The various servers involved had no problem carrying the loads.

Some makes/models of web-capable wireless devices did not process the HTML refresh scheme correctly. It appeared that some wireless carriers were prohibiting these refreshes as a bandwidth restriction measure. As more and more cellular carriers discontinue unlimited data plans, we may want to review the whole concept of automatic map refreshes in the future, and turn the associated decisions over to the individual end user. Or perhaps provide one map version with automatic refresh, and another with only user initiated manual refresh.

One advantage of the static base map design is that it is unlikely to change over the next few years. There may be slight changes in the various overlays, but these would likely be minimal. If this project is continued, future development and software coding workload should be modest.

5. Numbered footnotes, Links, and References:

[1] The Athens Ohio Marathon Home page:
http://athensmarathon.org/

[2] Bob Bruninga's (WB4APR) tutorial on 'ray tracing' in Google Earth:
http://aprs.org/hamtrails/aprsGoogleEarth.txt

[3] The Appalachian Trail Golden Packet Event (courtesy WB4APR):
http://aprs.org/at-golden-packet.html

[4] The KG6DI high-profile on-route digipeater (pictures and narrative):
http://w8kvk.com/KG6DI

[5] The Byonics Micro-Trak 'All-in-one' trackers used for this event:
http://www.byonics.com/microtrak/mtaio.php

[6] The lead runner bicycle mobile tracking station (courtesy WD8RIF):
http://home.frognet.net/~mcfadden/wd8rif/bicycle-mobile.htm

[7] The WIDEn-N APRS Network Protocol (courtesy WB4APR):
http://aprs.org/fix14439.html

[8] Using the APRS.fi Application Programming Interface (API):
http://aprs.fi/page/api

[9] APRS.fi is an APRS Mapping and Data internet service in Helsinki, Finland
http://www.aprs.fi/

[10] Using the Google Maps Application Programming Interface (API):
http://code.google.com/apis/maps/index.html

[11] Rails-to-Trails Conservancy Teams with Google for Bikeway Directions and Mapping:
http://www.railstotrails.org/news/newsroom/pressReleases/archives/20100310_DC_RTC_Google_Bike_Directions.html

6. Contacting the authors:

Ted Jacobson (W8KVK)
Athens, Ohio

Rob Jacobson (KB9AFT)
Sheboygan, Wisconsin

e-mail address for this project: dcc2011@w8kvk.com