AUTOMATIC POSITION REPORTING SYSTEM

The APRS Working Group

29 August 2000

Document Covers

Cover One

AUTOMATIC POSITION REPORTING SYSTEM

APRS PROTOCOL REFERENCE
Protocol Version 1.0

Authors The APRS Working Group
Document Version Approved Version 1.0.1
Filename aprs101.pdf
Date of Issue 29 August 2000

Technical Editor Ian Wade, G3NRW

Cover Two

Cover Three

APRS Protocol Reference
Protocol Version 1.0
by the APRS Working Group
Edited by Ian Wade

Published by
Tucson Amateur Packet Radio Corp
8987{309 East Tanque Verde Road, #337
Tucson, AZ 85749-9399
United States of America
http://www.tapr.org
ISBN 0-9644707-6-4
TAPR Publication Number: 99-4

Note on Reformatted Version

To be added

Prelude

FOREWORD

This APRS Protocol Reference document represents the coming-of-age of WB4APR's baby. Starting with a simple concept — a way to track the location of moving objects via packet radio — programs using the APRS protocol have grown into perhaps the most popular packet radio application in use today. It's also become one of the most complex; from the simple idea grew, and still grows, a tactical communications system of tremendous capability. Like many ham projects, the APRS protocol was designed as it was being implemented, and many of its intricacies have never been documented.

Until now. This specification defines the APRS on-air protocol with a precision and clarity that make it a model for future efforts. The work done by members of the APRS Working Group, as well as Technical Editor Ian Wade, G3NRW, should be recognized as a tremendous contribution to the packet radio art. With this document available, there is now no excuse for any developer to improperly implement the APRS protocol.

As an APRS Working Group member whose role was mainly that of observer, I was fascinated with the interplay among the APRS authors and the Technical Editor as the specification took form. Putting onto paper details that previously existed only in the minds of the authors exposed ambiguities, unconsidered consequences, and even errors in what the authors thought they knew. The discussion that followed each draft, and the questions Ian posed as he tried to wring out the uncertainties, gave everyone a better understanding of the protocol. I am sure that this process has already contributed to better interoperability among existing APRS applications. Everyone who has watched the specification develop, from the initial mention in April 1999 until release of this Version 1.0 document in August 2000, knows that the process took much longer than was hoped. At the same time, they saw the draft transformed from a skeleton into a hefty book of over 110 pages. With the specification now in hand, I think we can all say the wait was worth it. Congratulations to the APRS Working Group and, in particular, to G3NRW, for a major contribution to the literature of packet radio.

`John Ackermann, N8UR`
`TAPR Vice President and APRS Working Group Administrative Chair`
`August 2000`

Preamble

APRS Working Group

The APRS Working Group is an unincorporated association whose members undertake to further the use and enhance the value of the APRS protocols by (a) publishing and maintaining a formal APRS Protocol Specification; (b) publishing validation tests and other tools to enable compliance with the Specification; (c) supporting an APRS Certification program; and (d) generally working to improve the capabilities of APRS within the amateur radio community.

Although the Working Group may receive support from TAPR and other organizations, it is an independent body and is not affiliated with any organization. The Group has no budget, collects no dues, and owns no assets. The current members of the APRS Working Group are:

- John Ackermann, N8UR Administrative Chair & TAPR Representative
- Bob Bruninga, WB4APR Technical Chair, founder of APRS
- Brent Hildebrand, KH2Z Author of APRS+SA
- Stan Horzepa, WA1LOU Secretary
- Mike Musick, N0QBF Author of pocketAPRS
- Keith Sproul, WU2Z Co-Author of WinAPRS/MacAPRS/X-APRS
- Mark Sproul, KB2ICI Co-Author of WinAPRS/MacAPRS/X-APRS

Acknowledgements

This document is the result of contributions from many people. It includes much of the material produced by individual members of the Working Group.

In addition, the paper on the Mic-E data format by Alan Crosswell, N2YGK, and Ron Parsons, W5RKN was a useful starting point for explaining the complications of this format.

Document Version Number

Except for the very first public draft release of the APRS Protocol Reference, the document version number is a 3-part number "P.p.D" (for an approved document release) or a 4-part number "P.p.Dd" (for a draft release):

Thus, for example:

- Document version number "1.2.3" refers to document release 3 covering APRS Protocol Version 1.2.
- Document version number "1.2.3c" is draft "c" of that document.

Release History

The release history for this document is listed in Appendix 7.

Document Conventions

This document uses the following conventions:

- `Courier font` ASCII characters in APRS data.
- ␣ ASCII space character.
- … (ellipsis) zero or more characters.
- /$ Symbol from Primary Symbol Table.
- \$ Symbol from Alternate Symbol Table.
- 0x hexadecimal (e.g. 0x1d).
- All callsigns are assumed to have SSID –0 unless otherwise specified.
- Yellow marker (appears as light gray background in hard copy). Marks text of interest — especially useful for highlighting single literal ASCII characters (e.g. ") where they appear in APRS data.

- Shaded areas in packet format diagrams are optional fields.

Feedback

Please address your feedback or other comments regarding this document to the TAPR aprsspec mail list.

To join the list, start at http://www.tapr.org and then follow the path Special Interest Groups ⇒ APRS Specification ⇒ Join APRS Spec Discussion List.

Authors' Foreword

This reference document describes what is known as APRS Protocol Version 1.0, and is essentially a description of how APRS operates today. It is intended primarily for the programmer who wishes to develop APRS compliant applications, but will also be of interest to the ordinary user who wants to know more about what goes on "under the hood". It is not intended, however, to be a dry-as-dust, pedantic, RFC-style programming specification, to be read and understood only by the Mr Spocks of this world. We have included many items of general information which, although strictly not part of the formal protocol description, provide a useful background on how APRS is actually used on the air, and how it is implemented in APRS software. We hope this will put APRS into perspective, will make the document more readable, and will not offend the purists too much.

It is important to realize how APRS originated, and to understand the design philosophy behind it. In particular, we feel strongly that APRS is, and should remain, a light-weight tactical system — almost anyone should be able to use it in temporary situations (such as emergencies or mobile work or weather watching) with the minimum of training and equipment.

This document is the result of inputs from many people, and collated and massaged by the APRS Working Group. Our sincere thanks go to everyone who has contributed in putting it together and getting it onto the street. If you discover any errors or omissions or misleading statements, please let us know — the best way to do this is via the TAPR aprsspec mailing list at www.tapr.org.

Finally, users throughout the world are continually coming up with new ideas and suggestions for extending and improving APRS. We welcome them. Again, the best way to discuss these is via the aprsspec list.

–The APRS Working Group, August 2000

Disclaimer

Like any navigation system, APRS is not infallible. No one should rely blindly on APRS for navigation, or in life-and-death situations. Similarly, this specification is not infallible.

The members of the APRS Working Group have done their best to define the APRS protocol, but this protocol description may contain errors, or there may be omissions. It is very likely that not all APRS implementations will fully or correctly implement this specification, either today or in the future.

We urge anyone using or writing a program that implements this specification to exercise caution and good judgement. The APRS Working Group and the specification's Editor disclaim all liability for injury to persons or property that may result from the use of this specification or software implementing it.

Part I.

The Structure of this Specification

This specification describes the overall requirements for developing software that complies with APRS Protocol Version 1.0. The information flow starts with the standard AX.25 UI-frame, and progresses downwards into more and more detail as the use of each field in the frame is explored. A key feature of the specification is the inclusion of dozens of detailed examples of typical APRS packets and related math computations. Here is an outline of the chapters:

Introduction to APRS – A brief background to APRS and a summary of its main features.

The APRS Design Philosophy – The fundamentals of APRS, highlighting its use as a real-time tactical communications tool, the timing of APRS transmissions and the use of generic digipeating.

APRS and AX.25 – A brief refresher on the structure of the AX.25 UI-frame, with particular reference to the special ways in which APRS uses the Destination and Source Address fields and the Information field.

APRS Data in the AX.25 Destination and Source Address Fields – Details of generic APRS callsigns and callsigns that specify display symbols and APRS software version numbers. Also a summary of how Mic-E encoded data is stored in the Destination Address field, and how the Source Address SSID can specify a display icon.

APRS Data in the AX.25 Information Field – Details of the principal constituents of APRS data that are stored in the Information field. Contains the APRS Data Type Identifiers table, and a summary of all the different types of data that the Information field can hold.

Time and Position Formats – Information on formats for timestamps, latitude, longitude, position ambiguity, Maidenhead locators, NMEA data and altitude.

APRS Data Extensions – Details of optional data extensions for station course/speed, wind speed/direction, power/height/gain, pre-calculated radio range, DF signal strength and Area Object descriptor.

Position and DF Report Data Formats – Full details of these report formats.

Compressed Position Report Data Formats – Full details of how station position and APRS data extensions are compressed into very short packets.

Mic-E Data Format – Mic-E encoding of station lat/long position, altitude, course, speed, Mic-E message code, telemetry data and APRS digipeater path into the AX.25 Destination Address and Information fields.

Object and Item Reports – Full information on how to set up APRS Objects and Items, and details of the encoding of Area Objects (circles, lines, ellipses etc).

Weather Reports – Full format details for weather reports from standalone (positionless) weather stations and for reports containing position information. Also details of storm data format.

Telemetry Data – A description of the MIM/KPC-3+ telemetry data format, with supporting information on how to tailor the interpretation of the raw data to individual circumstances.

Messages, Bulletins and Announcements – Full format information.

Station Capabilities, Queries and Responses – Details of the ten different types of query and expected responses.

Status Reports – The format of general status messages, plus the special cases of using a status report to contain meteor scatter beam heading/power and Maidenhead locator.

Network Tunneling – The use of the Source Path Header to allow tunneling of APRS packets through third-party networks that do not understand AX.25 addresses, and the use of the third-party Data Type Identifier.

User-Defined Data Format – APRS allows users to define their own data formats for special purposes. This chapter describes how to do this.

Other Packets – A general statement on how APRS is to handle any other packet types that are not covered by this specification.

APRS Symbols – How to specify APRS symbols and symbol overlays, in position reports and in generic GPS destination callsigns.

APRS Data Formats – An appendix containing all the APRS data formats collected together for easy reference.

The APRS Symbol Tables – A complete listing of all the symbols in the Primary and Alternate Symbol Tables.

ASCII Code Table – The full ASCII code, including decimal and hex codes for each character (the decimal code is needed for compressed lat/long and altitude computations), together with the hex codes for bit-shifted ASCII characters in AX.25 addresses (useful for Mic-E decoding and general on-air packet monitoring).

Glossary – A handy one-stop reference for the many APRS-specific terms used in this specification.

References – Pointers to other documents that are relevant to this specification.

1. Chapter 1: Introduction to APRS

1.1. What is APRS?

APRS is short for Automatic Position Reporting System, which was designed by Bob Bruninga, WB4APR, and introduced by him at the 1992 TAPR/ ARRL Digital Communications Conference.

Fundamentally, APRS is a packet communications protocol for disseminating live data to everyone on a network in real time. Its most visual feature is the combination of packet radio with the Global Positioning System (GPS) satellite network, enabling radio amateurs to automatically display the positions of radio stations and other objects on maps on a PC. Other features not directly related to position reporting are supported, such as weather station reporting, direction finding and messaging.

APRS is different from regular packet in several ways:

- It provides maps and other data displays, for vehicle/personnel location and weather reporting in real time.
- It performs all communications using a one-to-many protocol, so that everyone is updated immediately.
- It uses generic digipeating, with well-known callsign aliases, so that prior knowledge of network topology is not required.
- It supports intelligent digipeating, with callsign substitution to reduce network flooding.
- Using AX.25 UI-frames, it supports two-way messaging and distribution of bulletins and announcements, leading to fast dissemination of text information.
- It supports communications with the Kenwood TH-D7 and TM-D700 radios, which have built-in TNC and APRS firmware.

Conventional packet radio is really only useful for passing bulk message traffic from point to point, and has traditionally been difficult to apply to real-time events where information has a very short lifetime. APRS turns packet radio into a real-time tactical communications and display system for emergencies and public service applications.

APRS provides universal connectivity to all stations, but avoids the complexity, time delays and limitations of a connected network. It permits any number of stations to exchange data just like voice users would on a voice net. Any station that has information to contribute simply sends it, and all stations receive it and log it.

APRS recognizes that one of the greatest real-time needs at any special event or emergency is the tracking of key assets. Where is the marathon leader? Where are the emergency vehicles? What's the weather at various points in the county? Where are the power lines down? Where is the head of the parade? Where is the mobile ATV camera? Where is the storm? To address these questions, APRS provides a fully featured automatic vehicle location and status reporting system. It can be used over any two-way radio system including amateur radio, marine band, and cellular phone. There is even an international live APRS tracking network on the Internet.

1.2. APRS Features

APRS runs on most platforms, including DOS, Windows 3.x, Windows 95/98, MacOS, Linux and Palm. Most implementations on these platforms support the main features of APRS:

- **Maps** –APRS station positions can be plotted in real-time on maps, with coverage from a few hundred yards to worldwide. Stations reporting a course and speed are dead-reckoned to their present position. Overlay databases of the locations of APRS digipeaters, US National Weather Service sites and even amateur radio stores are available. It is possible to zoom in to any point on the globe.

- **Weather Station Reporting** – APRS supports the automatic display of remote weather station information on the screen.

- **DX Cluster Reporting** – APRS an ideal tool for the DX cluster user. Small numbers of APRS stations connected to DX clusters can relay DX station information to many other stations in the local area, reducing overall packet load on the clusters.

- **Internet Access** – The Internet can be used transparently to cross-link local radio nets anywhere on the globe. It is possible to telnet into Internet APRS servers and see hundreds of stations from all over the world live. Everyone connected can feed their locally heard packets into the APRS server system and everyone everywhere can see them.

- **Messages** – Messages are two-way messages with acknowledgement. All incoming messages alert the user on arrival and are held on the message screen until killed.

- **Bulletins and Announcements** – Bulletins and announcements are addressed to everyone. Bulletins are sent a few times an hour for a few hours, and announcements less frequently but possibly over a few days.

- **Fixed Station Tracking** – In addition to automatically tracking mobile GPS/LORAN-equipped stations, APRS also tracks from manual reports or grid squares.

- **Objects** – Any user can place an APRS Object on his own map, and within seconds that object appears on all other station displays. This is particularly useful for tracking assets or people that are not equipped with trackers. Only one packet operator needs to know where things are (e.g. by monitoring voice traffic), and as he maintains the positions and movements of assets on his screen, all other stations running APRS will display the same information.

2. Chapter 2: APRS Design Philosophy

2.1. Net Cycle Time

It is important to note that APRS is primarily a real-time, tactical communications tool, to help the flow of information for things like special events, emergencies, Skywarn, the Emergency Operations Center and just plain in-the-field use under stress. But like the real world, for 99% of the time it is operating routinely, waiting for the unlikely serious event to happen.

Anything which is done to enhance APRS must not undermine its ability to operate in local areas under stress. Here are the details of that philosophy:

1. APRS uses the concept of a "net cycle time". This is the time within which a user should be able to hear (at least once) all APRS stations within range, to obtain a more or less complete picture of APRS activity. The net cycle time will vary according to local conditions and with the number of digipeaters through which APRS data travels.

2. The objective is to have a net cycle time of 10 minutes for local use. This means that within 10 minutes of arrival on the scene, it is possible to captured the entire tactical picture.

3. All stations, even fixed stations, should beacon their position at the net cycle time rate. In a stress situation, stations are coming and going all the time. The position reports show not only where stations are without asking, but also that they are still active.

4. It is not reasonable to assume that all APRS users responding to a stress event understand the ramifications of APRS and the statistics of the channel — user settings cannot be relied on to avoid killing a stressed net. Thus, to try to anticipate when the channel is under stress, APRS automatically adjusts its net cycle time according to the number of digipeaters in the UNPROTO path:

    - Direct operation (no digipeaters): 10 minutes (probably an event).

- Via one digipeater hop: 10 minutes (probably an event).
- Via two digipeater hops: 20 minutes.
- Via three or more digipeater hops: 30 minutes.

5. Since almost all home stations set their paths to three or more digipeaters, the default net cycle time for routine daily operation is 30 minutes. This should be a universal standard that everyone can bank on – if you routinely turn on your radio and APRS and do nothing else, then in 30 minutes you should have virtually the total picture of all APRS stations within range.

6. Since knowing where the digipeaters are located is fundamental to APRS connectivity, digipeaters should use multiple beacon commands to transmit position reports at different rates over different paths; i.e. every 10 minutes for sending position reports locally, and every 30 minutes for sending them via three digipeaters (plus others rates and distances as needed).

7. If the net cycle time is too long, users will be tempted to send queries for APRS stations. This will increase the traffic on the channel unnecessarily. Thus the recommended extremes for net cycle time are 10 and 30 minutes — this gives network designers the fundamental assumptions for channel loading necessary for good engineering design.

2.2. Packet Timing

Since APRS packets are error-free, but are not guaranteed delivery, APRS transmits information redundantly. To assure rapid delivery of new or changing data, and to preserve channel capacity by reducing interference from old data, APRS should transmit new information more frequently than old information.

There are several algorithms in use to achieve this:

- **Decay Algorithm** – Transmit a new packet once and n seconds later. Double the value of n for each new transmission. When n reaches the net cycle time, continue at that rate. Other factors besides "doubling" may be appropriate, such as for new message lines.

- **Fixed Rate** – Transmit a new packet once and n seconds later. Transmit it x times and stop.

- **Message-on-Heard** – Transmit a new packet according to either algorithm above. If the packet is still valid, and has not been acknowledged, and the net cycle time has been reached, then the recipient is probably not available. However, if a packet is then subsequently heard from the recipient, try once again to transmit the packet.

- **Time-Out** – This term is used to describe a time period beyond which it is reasonable to assume that a station no longer exists or is off the air if no packets have been heard from it. A period of 2 hours is suggested as the nominal default timeout. This time-out is not used in any transmitting algorithms, but is useful in some programs to decide when to cease displaying stations as "active". Note that on HF, signals come and go, so decisions about activity may need to be more flexible.

2.3. Generic Digipeating

The power of APRS in the field derives from the use of generic digipeating, in that packets are propagated without a priori knowledge of the network. There are six powerful techniques which have evolved since APRS was introduced in 1992:

1. **RELAY** – Every VHF APRS TNC is assumed to have an alias of RELAY, so that anyone can use it as a digipeater at any time.

2. **ECHO** – HF stations use the alias of ECHO as an alternative to RELAY. (However, bearing in mind the nature of HF propagation, this has the potential of causing interference over a wide area, and should only be used sparingly by mobile stations).

3. **WIDE** – Every high-site digipeater is assumed to have an alias of WIDE for longer distance communications.

4. **TRACE** – Every high-site digipeater that is using callsign substitution is assumed to have the alias of TRACE. These digipeaters self-identify packets they digipeat by inserting their own call in place of RELAY, WIDE or TRACE.

5. **WIDEn-N** – A digipeater that supports WIDEn-N digipeating will digipeat any WIDEn-N packet that is "new" and will subtract 1 from the SSID until the SSID reaches –0. The digipeater keeps a copy or a checksum of the packet and will not digipeat that packet again within (typically) 28 seconds. This considerably reduces the number of superfluous digipeats in areas with many digipeaters in radio range of each other.

6. **GATE** – This generic callsign is used by HF-to-VHF Gateway digipeaters. Any packet heard on HF via GATE will be digipeated locally on VHF. This permits local networks to keep an eye on the national and international picture.

2.4. Communicating Map Views Unambiguously

APRS is a tactical geographical system. To maximize its operational effectiveness and minimize confusion between operators of different systems, users need to have an unambiguous way to communicate to others the "location" and "size" (or area of coverage) of any map view. The APRS convention is by reference to a center and range which specify the geographical center and approximate radius of a circle that will fit in the map view independent of aspect ratio. The radius of the circle (in nautical miles, statute miles or km) is known as the "range scale". This convention gives all users a simple common basis for describing any specific map view to others over any communications medium or program.

3. Chapter 3: APRS and AX.25