Telpac and Paclink- Streamlined AX.25 Packet Server and Client for a full service Ham Radio Messaging Network

Rick Muething, KN6KB KN6KB@winlink.org
Vic Poor, W5SMM W5SMM@winlink.org

Abstract:
Ham radio's love affair with surplus hardware, budget software and a healthy volunteer spirit has served us well by keeping our hobby affordable while fostering many significant innovations. But the reality of high quality modern software is that it takes real programming skills and significant development time to build a worthwhile and reliable program. On-going enhancements and support also pose tough challenges when hams have come to expect free cradle-to-grave program maintenance. To manage development and maintenance efforts, a new approach had to be found for writing next generation packet servers and clients. These programs must be easy for sysops and users to setup, provide intuitive familiar user interfaces (e.g. Outlook, Netscape, Eudora) and reliably support both new and legacy BBS message systems and TNCs. This paper describes two examples of modern amateur packet programs along with implementation approaches that minimized the development effort and provided solid foundations for future contributions and easy maintenance.

Key Words:
BBS, Telpac, Paclink, FBB, Winlink 2000, WL2K, AGW Packet Engine, AX.25, MS Outlook, Eudora, Netscape, Telnet, MS VB.NET, Radio Addressing Schemes, Packet, Pactor, Emergency Communications.

Motivation for Telpac and Paclink:
The Winlink 2000 (WL2K) ham message system [1] has been very successful and enthusiastically adopted by mobile ham users particularly cruisers and RVers. The 4300 active users of Winlink 2000 now average over 5000 daily messages to both radio and Internet users. As awareness of WL2K spread, we were approached by a number of hams and emergency communication groups who wanted to expand the system's local area VHF/UHF coverage and further improve WL2K's suitability for emergency operations. But there were two obstacles to meeting those needs:

1) The complexity, computer requirements and equipment costs of configuring and maintaining a full service WL2K PMBO station.
2) The availability of a modern packet client program that could work with virtually all TNCs (including low-cost sound card and Baycom varieties) across all popular BBS systems. The user interface for this client had to be kept simple and familiar to allow non-radio savvy personnel to prepare and send messages during emergency operations.

Solving the first problem required dramatically simplifying the PMBO software and reducing or eliminating the large dynamic database that had to remain synchronized. We also had to find a way to support low cost and sound card TNCs with reliable, tested drivers while minimizing development and testing effort.

Jim Corenman, KE6RK, has written a very capable program called AirMail [2] that is the client of choice for HF Winlink users and includes packet support for a few dual port modems like the PTC II and KAM+. However, AirMail is used on both ham and commercial systems and Jim understandably has had to focus most of his support on the HF capabilities of AirMail. In addition, new demands are being made of packet client programs to simultaneously support multiple channels and work seamlessly with the nodes and switches popular in today's packet networks.

The two programs described here are what we developed to overcome these obstacles. Telpac establishes an easy-to-set-up access point that creates a bridge from the packet AX.25 world directly to a remote WL2K Telnet Server via Internet or wireless TCP connection. Paclink is a modern user client program that interfaces with familiar E-mail client programs and captures multiple TNC support by using the AGW Packet Engine [3].

The Development Philosophy

The first basic rule of developing complex programs and communication protocols is "don't reinvent the wheel" and that certainly holds true in this case. Luckily, many of the necessary components already existed and used standard, documented interfaces and protocols. One significant challenge in developing a client program is the user interface and the associated message composition, display, and management tasks. These functions, however, are exactly what are provided in most modern E-mail clients (Outlook/Outlook Express, Netscape, and Eudora). By disciplining ourselves to use existing protocol standards (POP3, SMTP, Telnet, TCPIP and BBS protocols), we significantly reduced the development effort and allowed the user to keep his familiar client interface. In addition, by interfacing both programs to the AGW Packet Engine we captured reliable drivers for most TNCs including low cost sound card and Baycom type modems. The AGW Packet Engine also allows multiple applications to share available TNC ports. We linked this development philosophy with a modern object-oriented language and development environment (VB.NET) to improve code reusability and further reduce the effort required to support future digital mechanisms.

Telpac . . . the TELnet PACket bridge

The Telpac program is an easily configured bridge that links AX.25 connections to a remote Winlink 2000 Telnet server. Since Telpac needs no local database or message routing, its internal structure is simplified and the required computer and memory resources are minimal. Telpac takes full advantage of WL2K's "*smart routing*", thus eliminating the need for users to declare a "home" BBS or having to worry about Hroute addressing. Mobile users simply connect to a Telpac station and exchange mail just as they do with any other Node of the WL2K system. Telpac supports the full capability of WL2K, including mixed radio and E-mail addresses, attachments, position reports and on-demand bulletins. Figure 1 shows how Telpac bridges the WL2K Telnet Server with packet radio client programs like Paclink or AirMail.

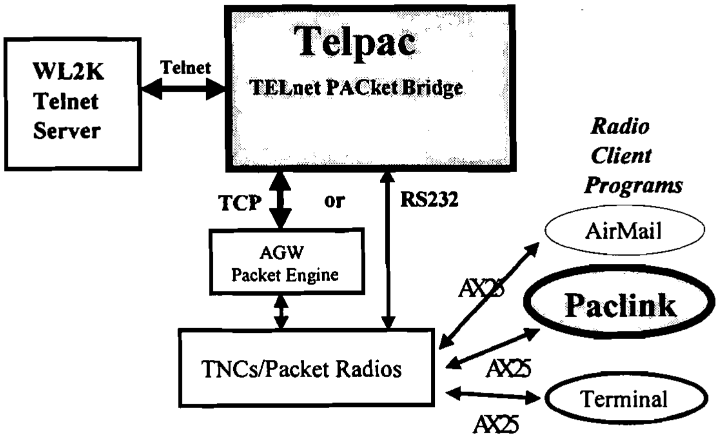


Figure 1. Telpac the TELnet PACket Bridge

Setting up Telpac is straightforward, with no routing tables, complex databases or configuration files to create or maintain. The sysop can use either direct RS232 control of popular TNCs or the more capable AGW Packet engine that supports most TNCs and permits sharing TNCs with other applications that use the packet engine. Telpac supports up to ten simultaneous connection streams on multiple radio ports and can interface to primary and backup remote WL2K Telnet servers with either a dial up or full time Internet connection. Since communication with the AGW Packet engine is via TCP, the Packet engine, TNC, and radios can be located on the same computer, at a remote site on the LAN or, in fact, anywhere on the Internet! Telpac supports conventional FBB BBS forwarding protocol and a simplified terminal interface mode. This allows the use of very simple terminals (e.g. PDAs or pocket PCs) with integrated Radio/TNCs (e.g. Kenwood TH-D7A) for compact portable client access.

Paclink . . . a multi-channel Personal BBS client

Paclink is a new implementation of a flexible, easily configured personal BBS (radio E-mail) client that interfaces with most popular ham BBS systems. Figure 2 shows the interfaces between the Paclink program and the other subsystems that complete the Personal BBS client.

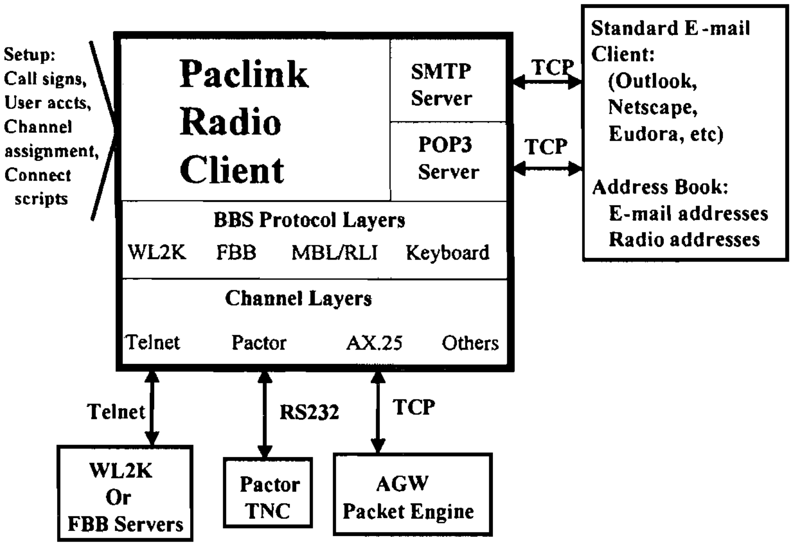


Figure 2. Paclink Radio Client and Subsystem Interfaces

It is worthwhile to review some of the key architectural features and tradeoffs that shaped Paclink. The first key implementation detail is that Paclink uses standardized TCP or Telnet protocols for communication between all major subsystems. The TCP connections are normally on the same computer, but could be located remotely on a LAN or accessed via the Internet. So, for example, if you are away from your station you could access your Paclink personal BBS over the Internet using any standard POP3/SMTP client with the correct user and password settings. This flexibility is what many emergency communication coordinators are now seeking. These flexible configurations allow logical and physical partitioning while still providing an acceptable level of security.

Another significant aspect of the Paclink architecture is its use of existing POP3 and SMTP compatible E-mail clients to provide much of the user interface, message addressing, composition, management and display functions. We used these familiar programs and their capabilities to support multiple mail accounts to leverage functionality and permit Paclink users to keep their familiar E-mail client for both radio and normal E-mail messages.

One of the important tradeoffs we made in specifying Paclink's functionality was not to implement full store-and-forward BBS capability. This functionality is not normally required in a Personal BBS and by eliminating it we relieved the burden of creating and maintaining complex routing tables.

Another important architectural decision was to require a one-to-one relationship between E-mail client "accounts" and connection "channels". Most E-mail programs support multiple E-mail accounts and these accounts, along with the corresponding setup in Paclink, specify everything necessary to establish the remote connection. This information identifies both outgoing and incoming connections and keeps data streams and messages separate from other connections or accounts. Figure 3 illustrates how this works in Paclink.

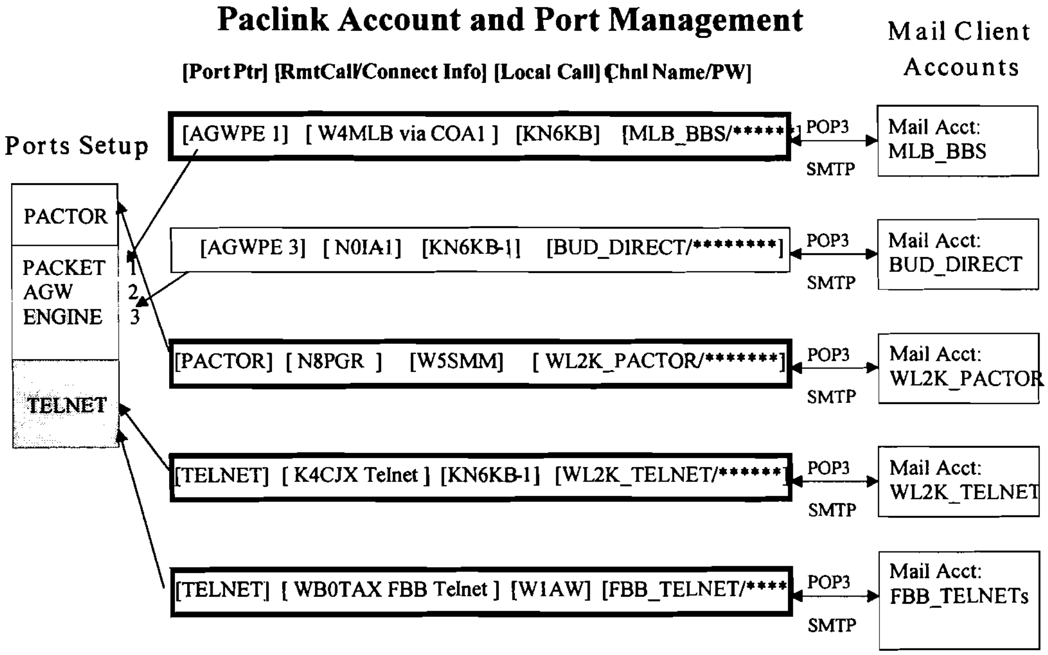
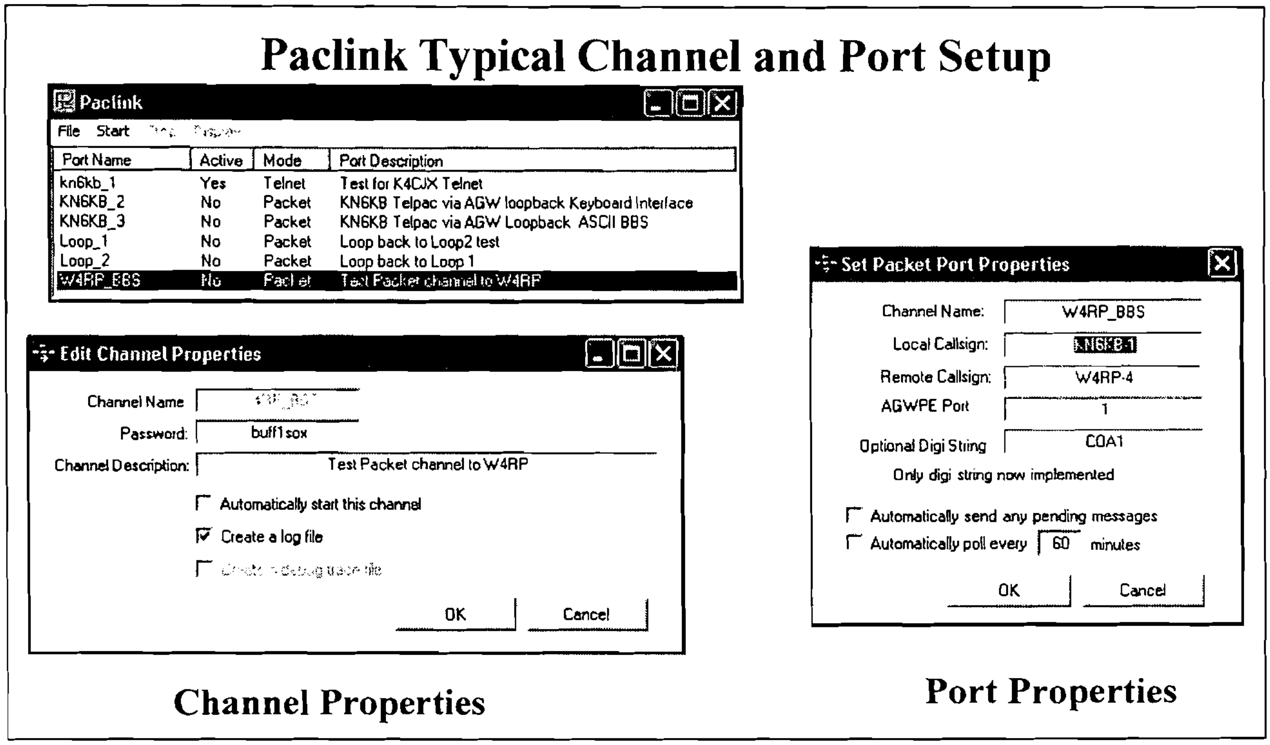

Paclink Account and Port Management
[Port Ptr] [RmtCall/Connect Info] [Local Call] [Chnl Name/PW]
Mail Client Accounts
Ports Setup
PACTOR
PACKET AGW ENGINE
1
2
3
TELNET
[AGWPE 1] [W4MLB via COA1] [KN6KB] [MLB_BBS/*****]
[AGWPE 3] [N0IA1] [KN6KB-1] [BUD_DIRECT/********]
[PACTOR] [N8PGR] [W5SMM] [WL2K_PACTOR/*******]
[TELNET] [K4CJX Telnet] [KN6KB-1] [WL2K_TELNET/******]
[TELNET] [WB0TAX FBB Telnet] [W1AW] [FBB_TELNET/****]
POP3
SMTP
Mail Acct: MLB_BBS
Mail Acct: BUD_DIRECT
Mail Acct: WL2K_PACTOR
Mail Acct: WL2K_TELNET
Mail Acct: FBB_TELNETs

Fig 3. Paclink's one-to-one relationship of channels to E-mail accounts

In figure 3 each unique E-mail account, as set up on the E-mail client, has a one-to-one association to a Paclink "channel" of the same name. This channel is then set up in Paclink to provide all necessary information to complete a radio or Telnet connection. Channel setup includes local and remote call signs, port numbers, protocols, polling interval and any optional connection script. Once these channels are configured in Paclink, all user interaction is normally done from the E-mail client program using appropriate E-mail or Radio address conventions. Fig 4 is a screen shot showing how a typical packet channel is configured in Paclink.



Paclink Typical Channel and Port Setup

Port Name	Active	Mode	Port Description
kn6kb_1	Yes	Telnet	Test for K4CJX Telnet
KN6KB_2	No	Packet	KN6KB Telpac via AGW loopback Keyboard Interface
KN6KB_3	No	Packet	KN6KB Telpac via AGW Loopback ASCII BBS
Loop_1	No	Packet	Loop back to Loop2 test
Loop_2	No	Packet	Loop back to Loop 1
W4RP_BBS	No	Packet	Test Packet channel to W4RP

Channel Name: W4RP_BBS

Password: buff1sox

Channel Description: Test Packet channel to W4RP

☐ Automatically start this channel

☑ Create a log file

☐ Create a debug trace file

Channel Name: W4RP_BBS

Local Callsign: KN6KB-1

Remote Callsign: W4RP-4

AGWPE Port: 1

Optional Digi String: COA1

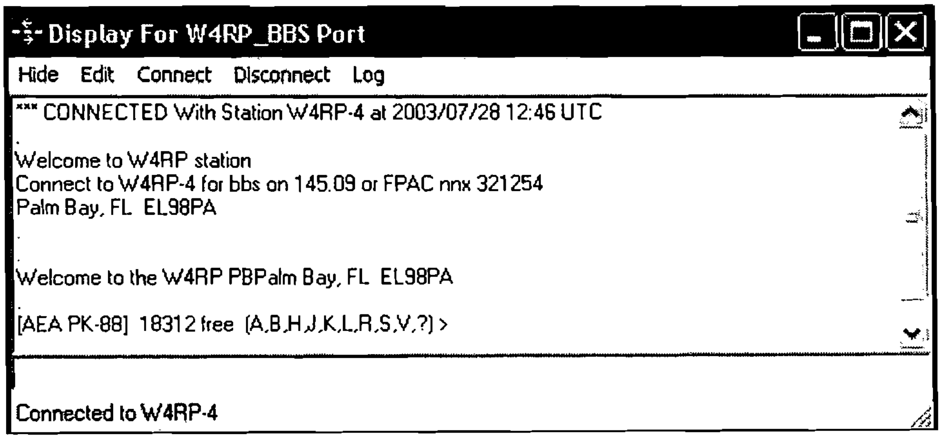
Only digi string now implemented

☐ Automatically send any pending messages

☐ Automatically poll every 60 minutes

Figure 4. Paclink channel and port properties for a Packet channel

After configuring each desired channel the only remaining setup required to operate Paclink is initializing the AGW Packet engine for your specific modem and radios and populating your favorite E-mail client with any required radio and E-mail addresses. We use a simple address syntax compatible with all E-mail clients to differentiate radio addresses from standard E-mail. This insures proper radio addressing including conventional packet Hroutes when required. Paclink automatically separates and duplicates messages to make them compatible with the target BBS for those systems or protocols that do not accommodate multiple addresses or mixed radio and E-mail addresses. In addition to its automated BBS <> BBS forwarding protocols, Paclink also supports classical terminal mode operation when necessary to work in conversational (keyboard) mode. Figure 5 shows Paclink's terminal display window while talking to a conventional BBS in keyboard mode.



```
*** CONNECTED With Station W4RP-4 at 2003/07/28 12:46 UTC

Welcome to W4RP station
Connect to W4RP-4 for bbs on 145.09 or FPAC nnx 321254
Palm Bay, FL  EL98PA


Welcome to the W4RP PBPalm Bay, FL  EL98PA

[AEA PK-88] 18312 free (A,B,H,J,K,L,R,S,V,?) >
```

```
Connected to W4RP-4
```

Figure 5. Typical Terminal Session with Conventional BBS (Keyboard mode)

Summary:

By carefully bounding the functionality of these programs and focusing on ease of setup and use, we have dramatically simplified the sysop and users efforts needed to expand existing BBS systems. We have also demonstrated how to rapidly develop reliable ham communication software that is both maintainable and expandable to meet future digital communication and message needs by using standardized protocols, interfaces, drivers and E-mail clients,

References:

[1] Winlink 2000...A Global Ham Message Transfer and Delivery Network;
Rick Muething, Vic Poor, Hans Kessler, and Steve Waterman. 19th ARRL and TAPR Digital Communications Conference, Sept 2000.
http://www.winlink.org/

[2] AirMail, a radio E-mail client program by Jim Corenman, KE6RK. http://www.airmail2000.com/

[3] The AGW Packet Engine by George Rossopoulos, SV2AGW
http://www.raag.org/sv2agw/pepro.htm