

# Sophisticated Mail User Interface Systems

**Keith Sproul, WU2Z**

**Mark Sproul, KB2ICI**

Radio Amateur Telecommunications Society

**Keith Sproul, WU2Z**  
Magnolia Road  
North Brunswick, NJ 08902  
AppleLink: Sproul.K  
Internet: wu2z@sproul.com

**Mark Sproul, KB2ICI**  
1368 Noah Road  
North Brunswick, NJ 08902  
AppleLink: Sproul.M  
Internet: sproul@sproul.com

## ABSTRACT

Most of us use many different electronic mail systems such as the Internet, Usenet, CompuServe™, America-Online™, Genie™, Prodigy™, and of course Amateur Radio Packet Mail. Most of these E-Mail systems have had some sort of sophisticated user interface developed for them. For Usenet, there are many different News Reader programs. CompuServe has CIS Navigator and CIS Information Manager. Many of us also use some sort of Local Area Network-based E-Mail systems at our work. LAN-based E-Mail systems such as Microsoft Mail™ are extremely sophisticated, especially those that run on windowing environments such as the Macintosh™ and Microsoft Windows?

With very few exceptions we all get paper mail delivered to our homes. We don't have to go to the post office to pick it up each day. The current packet mail systems require that we go to the post office (BBS) each day and ask for our mail. We also must ask for a list of public messages and explicitly specify which ones we 'want to see. Then we wait for 1200 baud data to come across the busy air waves.

A sophisticated user interface that would take care of retrieving a user's messages and specified bulletin topics would be a significant improvement. Such a system should deliver the mail that a user wants to his desktop, without the user having to go and get it manually. Such a system may also be expanded to then forward the mail to a user via another mail system so that the individual uses only one program at his computer.

The authors have developed a Packet Mail reader system that implements many of the sophisticated features seen in other mail systems. This same program can also gateway mail between Packet Mail and Microsoft Mail for the Macintosh.

## INTRODUCTION

Electronic mail systems on Local Area Networks (LANs) have evolved to the point where they are extremely robust and have many advanced features. These E-

mail systems also usually have a quite advanced user interface, especially the ones that run on the Macintosh, MS-DOS and Windows and other windowing environments. There are quite advanced

news reader systems available for reading the bulletins on the Internet. CompuServe has sophisticated software for accessing its services. Many other mail systems have fancy software to make reading mail easier. The Amateur Radio Packet community needs software with this type of LOCAL INTELLIGENCE to make the process of getting and reading one's mail easier.

With fancy computers on our desks and the availability of sophisticated programs and networks, we should be able to have systems where our electronic mail comes to us. This is the way it is on many LAN based E-Mail systems. If you use CompuServe and the CompuServe Information Navigator, you also get this kind of luxury.

### **BRIEF HISTORY OF AUTOMATIC SYSTEMS FOR PACKET RADIO**

In the early days of Packet Mail, the ONLY way to read messages and bulletins was to log onto your local PBBS and manually read them. This, unfortunately was (and still is) a slow process, especially at 1200 baud, and it is even worse if you are on a busy channel. Users wanted an easier way to read their packet mail.

Lan-Link for the PC has the capability to watch your local PBBS, detect if you have mail, connect to it when it sees mail for you, download the mail, and then disconnect. The messages are stored locally for you to access later at your convenience. Lan-Link can also read bulletins, and has a scripting system so that the user can customize it to his tastes and to be compatible with the PBBS system he is using.

The current generation of Packet Radio TNC's have 'mini' PBBS's built right into the TNC. PacComm calls theirs a PMS, (Personal Mail System), AEA calls theirs an MBX, (personal Mail Box), and the other manufacturers have similar features. These mini-PBBS's allow your local PBBS to forward your mail to you even when you are not there. Some sysops allow this and others do not,

Some people are also running stripped down versions of standard PBBS software, such as PRMBS and others as a 'personal' PBBS so that they can have their mail automatically forwarded to them.

All of the above solutions work and get the job done, some better than others. These types of solutions have their place, but most are designed just for personal messages and cannot handle large amounts of traffic.

The personal mail boxes work great if you and a few friends want to exchange mail but do not have good access to a full-function Packet BBS. This has the problem that they will only hold a very small amount of messages, therefore, are usually restricted to personal messages and not bulletins. This solution is generally used in less populated areas where full-function PBBS's aren't very common.

Lan-Link and the stripped-down versions of the PBBS systems solve the problem of not having enough space but require the full time use of the computer and do not allow anything else to be done at the same time.

The big problem with all of the above solutions is that you still end up with the same old command-line interface. This leaves the user with having to look through a single list of messages that were received

by his 'personal BBS system'. He still has to decide what to read and what not to. With some of the stripped-down PBBS systems, you can program them to select what types of bulletins you want to receive and what not to receive. Even so, the user still has a single list of messages to read through.

### ADVANTAGES OF AUTOMATIC SYSTEMS

The main advantage to this type of system is that your computer can go get your messages for you. Therefore, this can happen when the frequency is less busy, or if it takes awhile, you are not sitting there waiting for it.

Since the sending and receiving is automatic, it can be scheduled to be done during non-prime time, i.e. late at night, or during normal work hours when there isn't much packet activity. This alone will improve the throughput for everybody, including the people that still read Packet Mail the old way because there will be less people on when they are on.

When you decide to read your messages, you can read them as fast as your computer can bring them up on your screen.

### DISADVANTAGES OF AUTOMATIC SYSTEMS

Most of the automatic systems need to be left on 24 hours a day and require dedicating a computer to them.

Unless you are only receiving messages addressed directly to you, the system will occasionally receive messages that you didn't really want to read, thus causing more traffic than there would have

been if you read your mail the old manual way.

If you are not careful, your hard disk will fill up very fast, especially when you are away from home: for awhile.

### INTELLIGENT AUTOMATIC SYSTEMS

We have already discussed Automatic systems above. I define an INTELLIGENT system as a system that can be tailored to get the types of messages that we want and not bother us with messages that we don't want. This has two main advantages. First, we are not bothered with things we are not interested in. Second, and more important, the airwaves are not wasted with the transfer of data that is not needed.

Given an automatic system for sending and receiving Packet Mail messages, you have the basis for developing a sophisticated user interface that can make the task of sorting through and reading the large amount of messages and bulletins significantly easier. With today's large volume of information, this type of electronic assistant is greatly needed. We already have an information overload which Packet bulletins are contributing to this also.

An intelligent system should be able to do many things for you. It should be able to

- **Sort** messages by topic.
- Scan subjects for keywords.
- Sort and select bulletins based on your personal preferences.
- Keep track of what messages you have read.
- Know about certain types of messages that are time critical.

- Update regularly sent messages such as satellite keplerian data sets.
- Keep track of the different people you send mail to so that you don't have to remember their full address every time you need to send them a message.
- Handle replying and forwarding of messages.
- Handle address lists and many other features that are common in LAN based E-Mail systems.

### DESIGN GOALS

The following features are available on the LAN-based E-Mail system I use at work. I would like a system that does this for Packet Mail too:

- Automatic marking of read messages.
- Automatic address book.
- Easy scheduling.
- Easy replies and forwarding.
- Able to be run in the background.
- Address Lists.
- Aliases.
- Immediate notification when new messages arrive.
- User specifications of sorting criteria.

For a Packet Mail system the following features would also be desirable:

- Automatic sorting of bulletins by topic.
- 24 hour operation is not required.
- Easy selection of accept/reject criteria, based on either the TO/Distribution field, and the FROM field.
- Automatic superseding of messages such as KEPS.
- Selective automatic deleting of dated messages.

Having a system that implements even a few of these features would be a good step in the right direction. Also, with some **PBBS's** having over a hundred new messages a day, any system that can intelligently reduce message traffic is a plus for the whole system.

### HISTORY OF OUR PROJECT

This project started out as a simple desire to write a gateway between Microsoft Mail for the Macintosh and the standard **WORLI/WA7MBL** compatible PBBS systems. After pursuing this for awhile and getting some of it working, it was determined that by the time this project was done, we would have a message handling program that would have lots of nice features. These features would also be useful to someone that did not have Microsoft Mail. At this point, we backtracked a little and started working on a mail reader program that would also have the ability to forward messages to Microsoft Mail. This was implemented as an option that could be enabled or disabled.

With this type of gateway capability, we started to look at many other options too. The software is implemented such that it would be relatively easy to add gateways to other E-Mail systems and have them enabled or disabled also.

### FUTURE CONSIDERATIONS FOR AUTOMATIC MAIL SYSTEMS

As discussed here, all of the above automatic mail systems work with existing PBBS protocols. The most common one in use today is the **WORLI/WA7MBL**. There is growing support for the FBB compression protocol. Unless somebody

comes out with a better compression standard, we expect to implement this into our software.

As these automatic message systems become more and more widespread, we are going to start to see many different problems.

The biggest single problem that needs to be addressed is the ability to have a MULTICAST message system. A multicast message system is where a single PBBS will transmit a bulletin or message and as many other PBBS's that can hear it will receive it at the same time. This topic has been discussed several times in the past. See the references, [1], [2], and [3].

Another feature needed by mail protocols are remotely expanding mail lists, i.e. RMailer in the PRMBS PBBS system. This is the topic of a paper by Frank Warren, KB4CYC, also in these proceedings.

We also need to look into either expanding our current mail protocols or adopting others so that the Packet Mail systems can expand with our growing needs. [4]. Other features that would be useful would be return-receipt, enclosures, expiration dates, message size, supersede flags, i.e. this messages supersedes a previous message, and many other features not currently available.

These types of features are needed in the mail-forwarding protocol because the software is going to very rapidly need them. If they are not there, there will be lots of unneeded traffic that these additions to the protocols could avoid.

### MACINTOSH PACKET MAIL PROTOCOLS

Mac Packet Mail, MPM, talks to other BBS's via the standard WORLI/ WA7MBL protocols. The information exchanged in the header information of this protocol is enough for a PBBS system to know if it wants a particular message BEFORE it has actually received the entire message. In the actual protocol, the sending PBBS sends the header information, which includes what type of message, private or bulletin, who it is to, who it is from, and a BID/MID number. The TO address is either an actual person, or a distribution such as ALLBBS, or 4SALE, etc. MPM has a list of what types of messages it wants to receive. It also has a list of what types to reject. This is needed because there will always be: people that address their messages incorrectly. Since the protocol allows for a message to be either rejected, received, or deferred, an intelligent system can also save the headers of other messages, and if the user wants to receive them, he can mark them for later retrieval. The system will get it the next time it logs on. This would be used for topics that the user usually doesn't want to read, but knows that once in a while he will want to. However, this would have to be set up ahead of time. Unfortunately, the subject is NOT transmitted with the header. It is not received unless the receiving BBS has said it wants the entire message.

### MACINTOSH PACKET MAIL USER INTERFACE

The user interface for Macintosh Packet Mail has many features that resemble current LAN-based E-Mail systems. This interface is still under development as we continue to work on

the system (See **Figure 1**). This interface is what the end user sees and this is what will make or break a system like this. The goal is to make it easy for the user to navigate through the large piles of mail and bulletins that come through the system every day.

Once a user has this set up, he should be able to read his mail in much less time, get more out of it, and not have to wait for slow channels. This will do a lot to help the distribution of the vast quantities of information we have to deal with every day.

### MACINTOSH PACKET MAIL MICROSOFT MAIL GATEWAY

Microsoft has a Software Development Kit available for developers that want to write gateways to Microsoft Mail. We purchased this gateway and used it to write the hooks into Microsoft Mail. The software supplied with this kit made developing a gateway into and out of Microsoft Mail easier than expected.

We had to develop our program so that it could read and write files that the Microsoft Mail gateway program could understand. This turned out to be pretty straightforward.

On the Macintosh, you can run multiple programs simultaneously, therefore, the Mail server and the Macintosh Packet Mail Program can run on the same computer. This allows one computer that is already running 24 hours a day to do the additional task of gatewaying between the LAN-based E-Mail system and Packet Mail.

Built into Microsoft Mail is an account/password system. The gateway system has an additional account system.

This turned out to be very useful in restricting non-Hams from using the gateway by accident? All that has to be done is to create a gateway account for each Ham on the mail system and name the account with the user's call letters. With this method, even though the user's Microsoft Mail account is his name, the gateway account can be his call letters. This lets the gateway do the call letters to user mapping without having to write any extra software.

If someone on the mail system tries to use the gateway, he will get a rejection message telling him that he is not authorized to use that gateway and to contact his network administrator.

With this method of mail gateways, a user can send a single message to multiple people on the Microsoft Mail server AND to other people on Packet Mail.

The gateway software provides for convenient mail addressing too. Also, once you entered the call letters and BBS name once, you won't ever have to enter it again. See Figure I for who the BBS addressing is done.

### CONCLUSIONS

With the onslaught of the information revolution, we are inundated with huge amounts of data. This is now starting to hit the Packet Mail systems. Therefore, to be able to keep up with what is going on, we need our computers to be able to do more work for us. Having a mail reader program that can make our reading of mail easier and quicker is very desirable.

In pursuing this type of sophisticated user interface, we discover that some things are not doable with the current protocols. Even though the message

## **Sophisticated Mail User Interface Systems**

**WU2Z & KB2IC**

transfer protocols that are used by the Packet Mail systems aren't very elaborate, it is still possible to develop a sophisticated package for the end user.

Figure 1

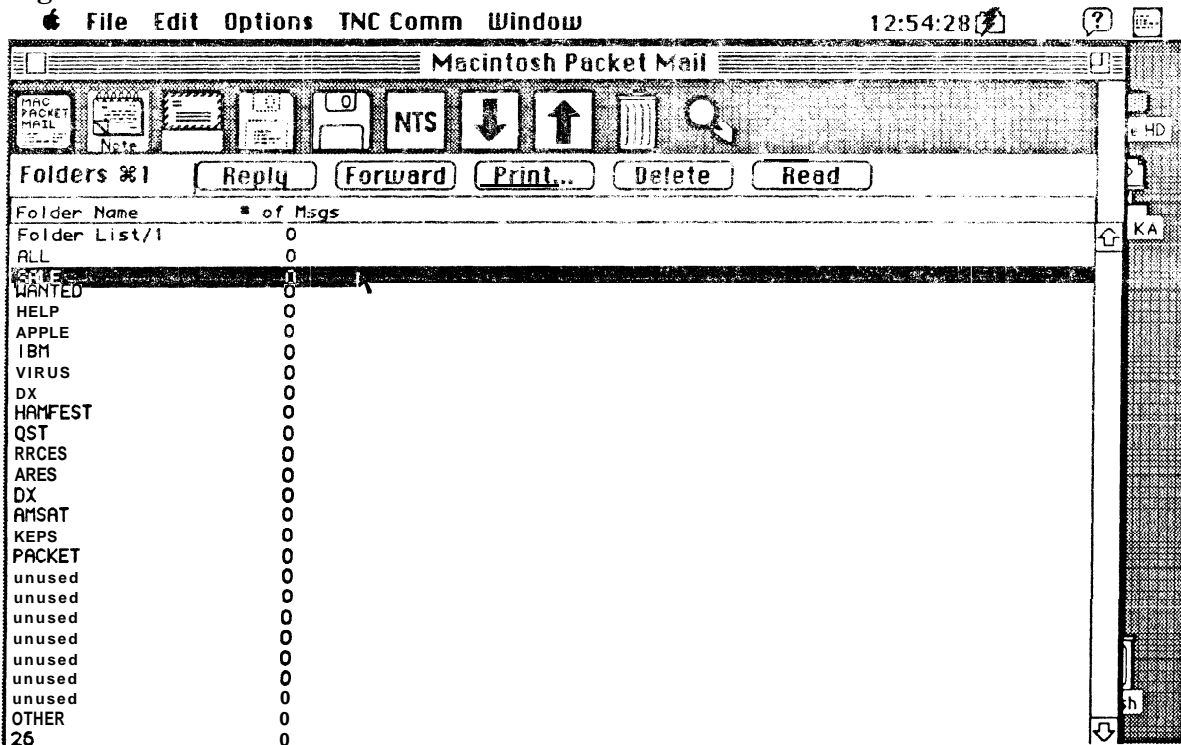
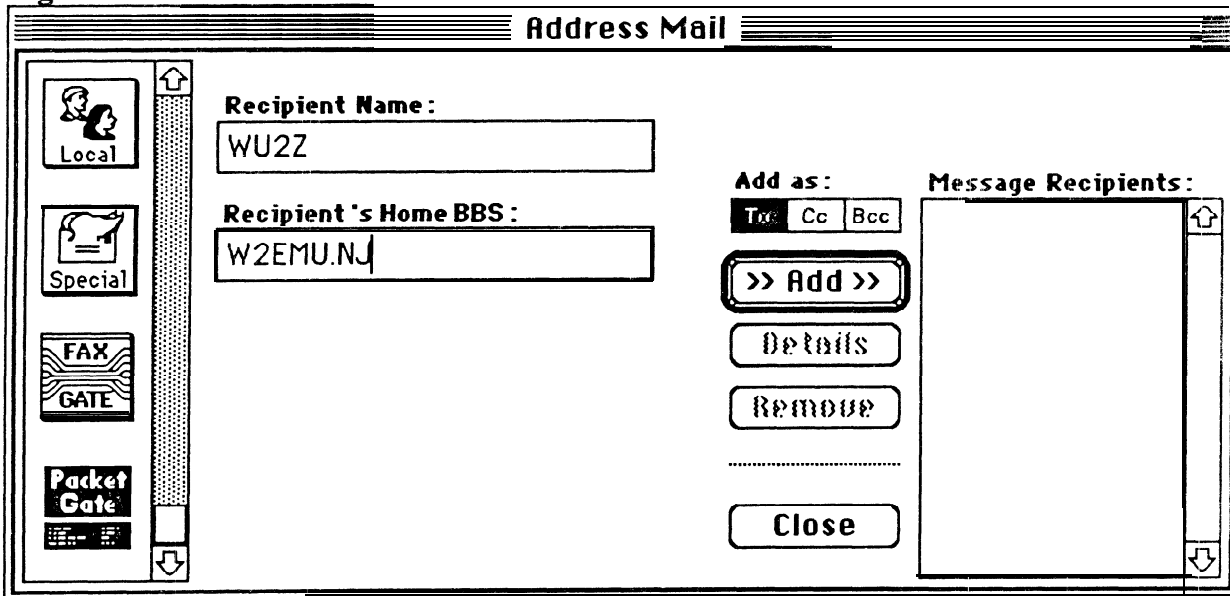


Figure 2





**REFERENCES**

[1] **Beattie, J. Gordon Jr., N2DSY** *A Packet Broadcast Protocol*, Proceedings of the 8th Computer Networking Conference, Newington, CT: ARRL, 1989 pp 1-15

[2] **Koopman, Derek, G1TLH** *Connectionless Mail Protocol and More*, QEX ARRL Experimenters' Exchange, Newington, CT; ARRL, July 1990, pp 19-22

[3] **Price, Harold E., NK6K and Ward, Jeff, G0/K8KA** *PACSAT Broadcast Protocol*, Proceedings of the 9th Computer Networking Conference, Newington, CT; ARRL, 1990, pp 232-244

[4] **Klemets, Anders, SMORGV** *The Network News Transfer Protocol*, Proceedings of the 9th Computer Networking Conference, Newington, CT; ARRL, 1990, pp 150-153