Messages - a Multi-Media Mailer

As published in the *Linux Journal*, Octoberr, 1994

In this article, Terry Gliedt continues his tour through the Andrew project - this time into a new area, multi-media mail.

Introduction

AUIS had its roots in 1982 when Carnegie Mellon University and the IBM Corporation decided to jointly develop a campus computing facility based on personal computers to replace the time-sharing system then on campus. IBM provided not only generous funding, but also some talented individuals and access to IBM development programs.

The result was a graphical user interface we know as the Andrew User Interface System and a file system, the *Andrew File System*. The file system formed the basis of Transarc Corporation's *Distributed File System* (DFS) and is as part of the Open System Foundation software.

The ***Andrew Consortium***, composed of a number of corporations and universities, funds the current development of AUIS. AUIS is available on a wide variety of platforms including Linux, AIX, Solaris, Ultrix, HP UX as well as others.

In early June, version 6.3 of the Andrew User Interface System (AUIS) was released by the Andrew Consortium. This led to the release of **auis63L?-wp.tgz** which contains just a small portion of AUIS that is suitable as a word processor.

Now another package has been released, **auis63L?-mail.tgz**, to *sunsite.unc.edu* in */pub/Linux/X11/andrew*. This package provides the Linux community a powerful mail system which leverages the power of the tools we see in the word processing package to provide multi-media mail.

Multi-Media Mail

Electronic mail is becoming enormously popular and common these days, but most of the tools we use have hardly changed since the earliest e-mail systems were invented. A major effort in the Andrew project was the development of the Andrew Mail System (AMS) and its mail reading tool, *messages*. The developers at CMU wanted a mailer which not only met the needs of traditional mail systems, but could also be used for both public and private bulletin boards. Today at CMU today one can follow over 7500 bulletin boards with *messages*. The scale of the data is a major factor in the design of *messages*.

Another goal for the developers of *messages* was to provide the multi-media mail system. The AUIS data-stream was designed from its inception to be mail-able. This means it uses only 7-bit ASCII data which isn't "too long" for the various mail systems one might encounter. From a technical standpoint, this design has worked well.

MIME

Early versions of *messages* suffered from the chicken-and-egg syndrome. Since there was no standard at the time, most people were reluctant to use *messages* because not enough other people used it. On the other hand, if a large number of people used it, then everyone would love to use messages. For instance, if telephones were very uncommon and only 1% of people had one, your business would be unlikely to install a telephone. On the other hand, no business today would think of not having a telephone, since virtually everyone has a telephone.

Everyone seemed to think *messages* was great - but at the time you could only send AUIS-mail to others who used *messages*. On large homogenous sites like CMU, this worked fine. But how to convince the rest of the world to convert?

After years of effort, the answer became obvious - you can't. Not everyone was going to convert to *messages* - no matter how wonderful it might have been. One of the principle architects of *messages*, Nathaniel Borenstein, decided to take another tack. He proposed an architecture called MIME (Multipurpose Internet Mail Extensions) that would allow one to send non-traditional mail (like pictures) in an architected format. Nathaniel retrofitted *messages* to send mail in MIME format instead of native Andrew Mail format. When receiving MIME mail, he modified *messages* to call a new program, *metamail*, to display non-text on your machine. After demonstating that this all worked, Nathaniel provided similar updates to several common mailers to support MIME in the same manner.

For example, if I send mail using *messages* with a picture to someone who uses some other mailer on a SUN workstation, the recipient's mailer might end up calling "xv" to display the picture. In a non-X environment, perhaps some other tool might be called. Finally, if the picture just cannot be displayed, then the mailer might "convert" the picture into some text like "a picture was provided here, but cannot be displayed".

Messages

Messages is actually just one part of a larger system known as the Andrew Mail System. This system supports reading and posting to bulletin boards and delivery between cells in the Andrew File System. I will not discuss any of these topics here, but rather just describe *messages* as a conventional mail user agent.

To view your mail, invoke the command *messages* from an *xterm* window. In Figure 1. you can see that the resulting *messages* window has three parts. The top window shows a list of folders where you can save mail. The folder "mail" is where your incoming mail is saved until you move it elsewhere or delete it. When you use *messages* the first time, you will only have a mail folder. Begin by selecting a folder (probably **mail**) by pointing to a name and clicking with the left mouse button.

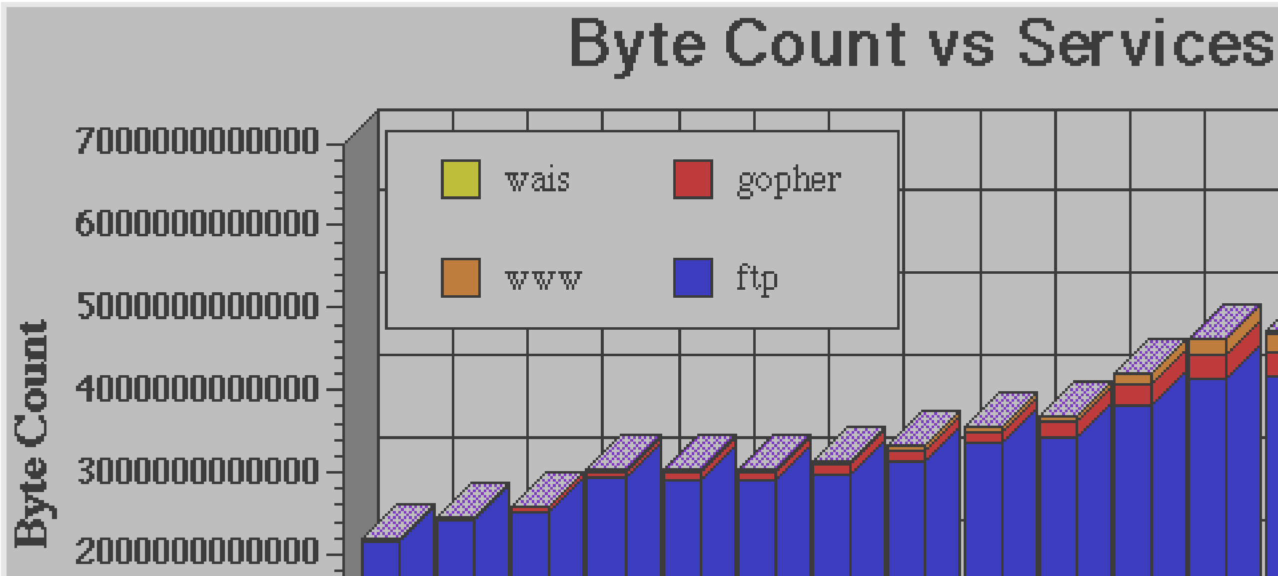
messages Search/Spell This Message File Into... Message Folders Other Send/File Marked More...

AUIS–old–mail
BlindCC
Linux
Linux–Journal
NeXT
Personal
mail (Mail ; All 10 shown)

5–Jun–94 *Re: What version am I using?* – sws@tora.RoBIN.de (1727)
6–Jun–94 *Re: Andrew Conference* – Susan Straub@andrew.cmu. (575)
8–Jun–94 *Re: Andrew Conference* – Susan Straub@andrew.cmu. (889+0)
9–Jun–94 *Re: Sound inset* – Todd Inglett@VNET.IBM.CO (352+1)
9–Jun–94 *RE: What version am I using?* – Oliver Daudey@backdoor.h (3380)
11–Jun–94 *NSFnet Backbone Statistics* – => tpg@trillian (60+1)
11–Jun–94 *Throw this away* – => tpg@trillian (7)

Date: Sat, 11 Jun 1994 11:05:20 –0500 (CDT)
From: Terry Gliedt <tpg@mr.net>
To: tpg@trillian
Subject: NSFnet Backbone Statistics

Here's what's going on with the **Information SuperHighway**.



The middle window will now display a summary of the mail in that folder. Select one of these and you'll see the summary line in bold and in the bottom window, the text of the message. In Figure 1. you will notice several things. The headers of the mail (From:, To:, etc.) are show in bold. When you delete a piece of mail, the summary line is shown in a smaller font, but the actual file is not deleted until you explicitly delete it. More discerning eyes will notice the scroll bar shows there is something above the **From:** line. This data is the normal mail headers which you normally do not want to see. If you look at the headers for this particular piece of mail, you'd see that it has the lines:

MIME-Version: 1.0
Content-Type: multipart/alternative;
boundary="Interpart.Boundary.ohyS2z9z0001I1RlEF"

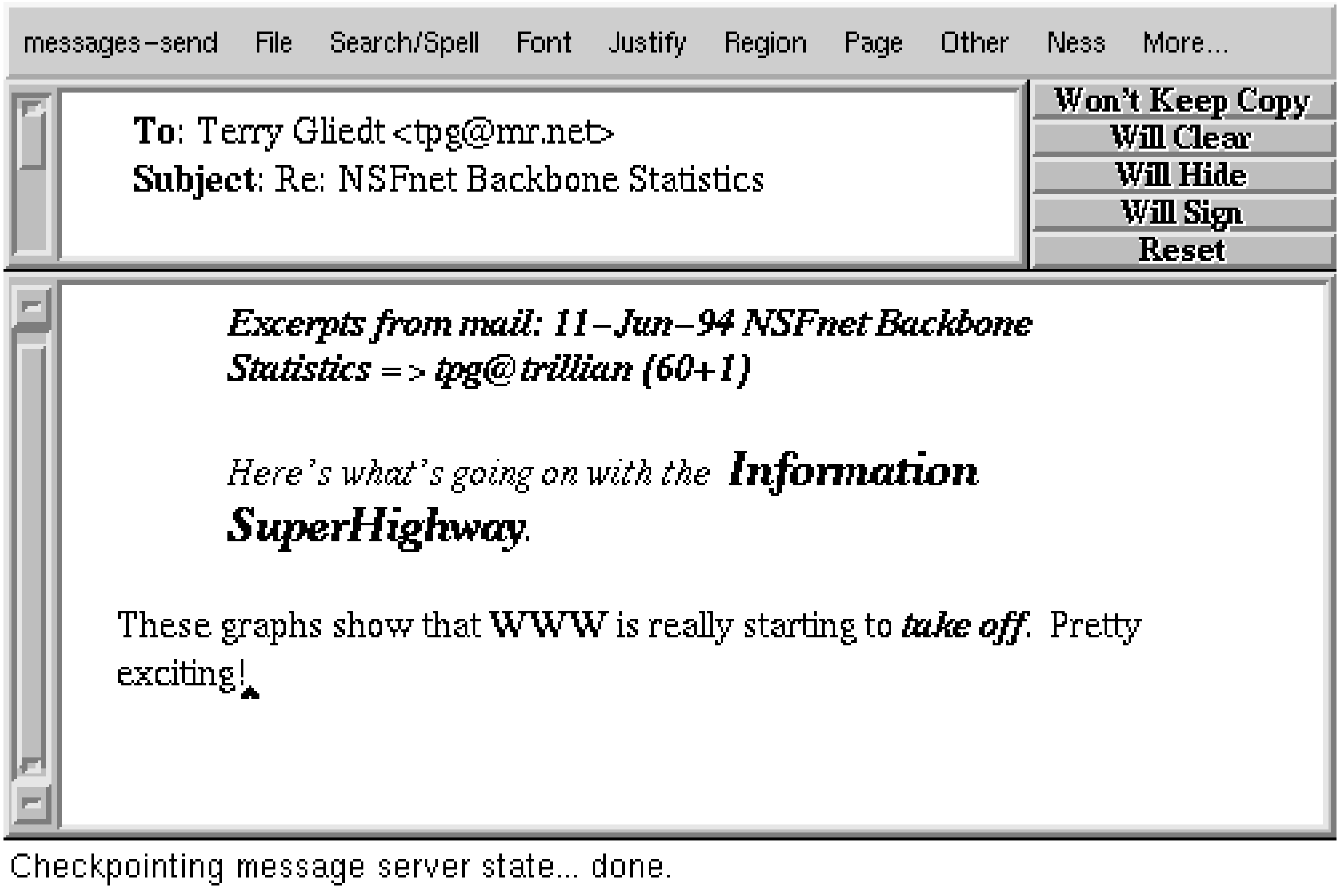
indicating this is MIME mail and consists of several parts. If you look at the summary line for this mail, you'll notice the phrase "(60+1)", meaning the body consists of 60 characters and contains one inset.

In the body of the mail, you can see the phrase "Information SuperHighway" is in a larger bold font. MIME's metamail or other low-end MIME reader would convert this to simple text on systems which do not have font capabilities. On some systems the picture might show up in a separate window. Since *messages* has all the capabilities of AUIS, the text and images are shown in-line, as they were intended.

Sending Mail

To compose and send mail to someone, select *Send/Post Message* on the *messages* menucard. This will present a window as shown in Figure 2. In many ways, this is just another *ez* session (*ez* is the basic editor in AUIS and was described in more detail in previous issues of LJ). In this particular example, I selected the mail we saw in Figure 1 and then I selected *Reply to Sender* on the *messages* menucard. This brought up the ***messages-send*** window with the **Subject:** and **To:** field already filled out. I then selected the first sentence of the original mail (so it was shown in reverse video) and then in the ***messages-send*** window, selected *Excerpt body* on the *Other* menucard. This resulted in the indented and italized *Excerpts* lines that you see.

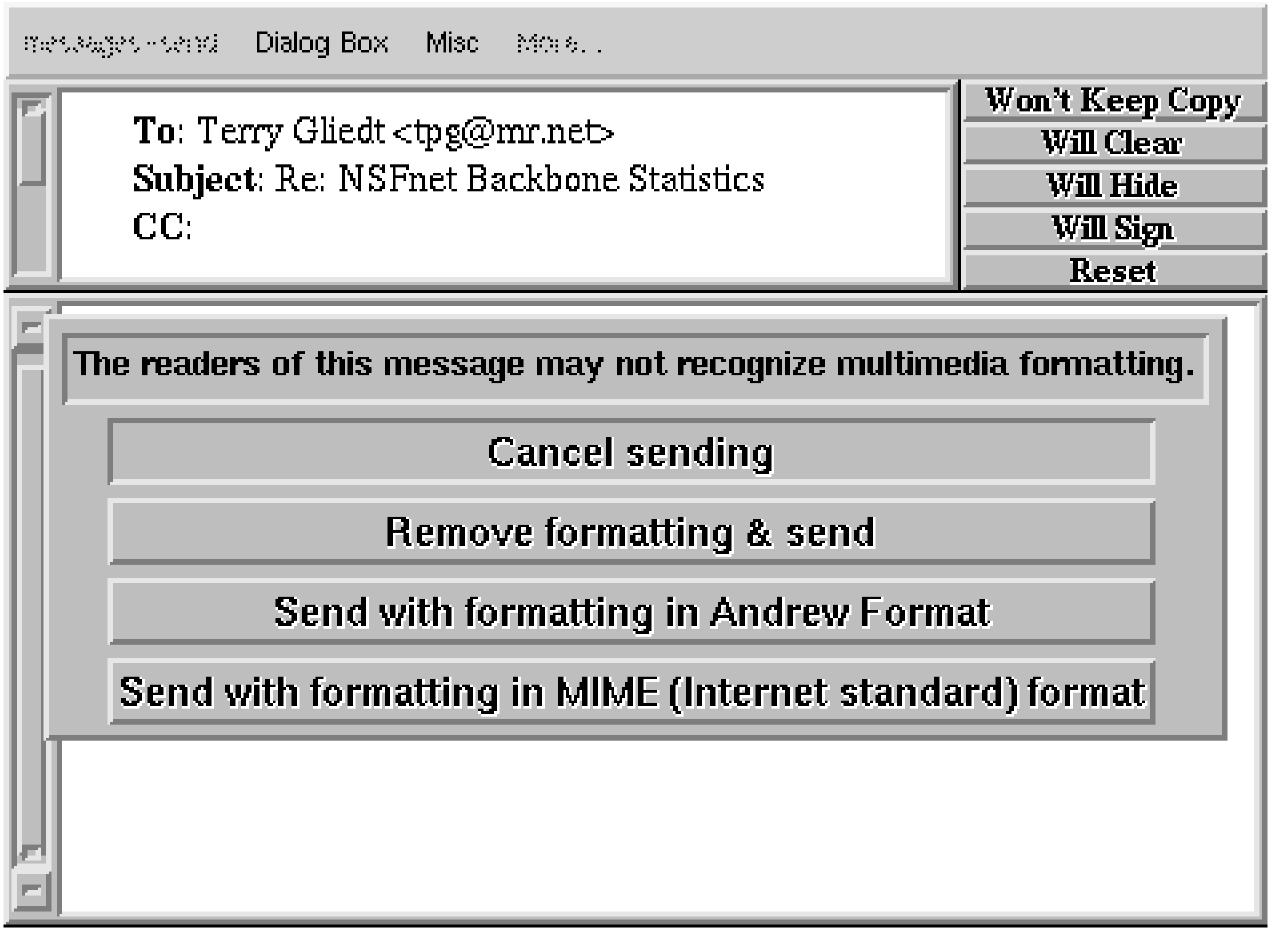
Had we not marked any text and simply selected this menu card, the entire original message body would have been copied, indented and italized.



Notice that the excerpted area may contain more than simple text, but also contains the large bolded **Information SuperHighway**. In fact we could have also included the image you saw in Figure 1. To send the mail, select *Send/Post* on the *messages-send* menucard. If the mail contains multi-media (anything except simple plain ASCII text), you will be prompted as shown in Figure 3. If you select *Remove formatting and send*, *messages* will do some simple text conversion. The excerpt will have the font information removed and the excerpt text itself will be prefixed with ">" as is common with many mail readers. As you can see in Figure 3., you may choose to send the mail in the original AUIS format or in MIME format. You can avoid the question altogether and always send in MIME format with an entry in your *$HOME/preferences* file like this:

```
*.mailsendingformat: mime
```

Issue the command **auishelp preferences** and read about the ***mailsendingformat*** setting.



Options

Everyone wants to do mail their own way - and the same is true with *messages*. *Messages* has dozens of options you may set - so many so, that it provides its own interface to setting and querying these. If you select *Set Options* on the *Other* menucard, *messages* will present a list of the options and their current setting. I'll bet as soon as you have *messages* working, it won't be long before you will be poring through the options, trying them out.

Compatibility with Other Mail Readers

The designers chose to not keep the *messages* database compatible with that of other Unix mail readers like *elm*. In conventional Un*x mail user agent (e.g. *mail*) when mail is received, it is first stored in /usr/spool/mail/$USER as a simple "flat" file. All the mail is mashed together in one physical file. It's up to the mail reader to sort these out. This works fine when the file is small, but when you have 100 pieces of multi-media mail, each 50K in size, it starts getting unwieldy and slow.

Now when you invoke your mail reader (e.g. *elm*), the reader shows you what is in /usr/spool/mail/$USER. When you save the mail to a folder, the mail is appended to some file in $HOME/Mail (e.g. ~/Mail/tpg in my case). Just as with the mail in /usr/spool/mail, this is also a simple file containing many logical files (pieces of mail). It will suffer even more from performance problems as you keep more and more mail around. Most mail readers allow you to create separate folders for categories of mail - but each still uses one monolithic file.

In the world where AUIS was developed, it is not unusual for one to have hundreds of pieces of mail (and large multi-media mail at that). So another, incompatible approach was taken. *Messages* keeps each individual piece of mail as a separate file in a folder (i.e. directory) and builds an index so it can quickly show whats in the folder. Each of these folders is kept in the directory $HOME/.MESSAGES.

The drawback to this approach is that you cannot directly switch between *elm*, for example, and *messages*. Any mail received by *messages* is not available in your next *elm* session. Similarly, mail you received in the past with *elm*, is not directly available in *messages*. However, that does not mean it cannot be done - but you must do it "manually". I have developed some techiques to allow you to "convert" your *elm* mail for use in *messages* and vice versa. I will not describe these in detail, but rather direct you to read about it in /usr/andrew/README.ez.mail which is created when you install the ***auis63L?-mail*** package.

Printing and Previewing

Just as AUIS prints other text documents, *messages* will print your mail using PostScript. In this version AUIS objects all generate *troff* output - along with copious amounts of embedded PostScript. The *troff* is then processed to generate the necessary PostScript. The default print command will invoke a shell, */usr/andrew/etc/atkprint*. The default preview command calls the shell */usr/andrew/etc/atkpreview*. Each of these shells will invoke the *groff* formatter to generate the PostScript output. In *atkpreview* the *groff* output is directed into *ghostview*.

For More Information

A mailing list is available at info-andrew@andrew.cmu.edu (mail to info-andrew-request@andrew.cmu.edu for subscriptions). The newsgroup comp.soft-sys.andrew is dedicated to the discussion of AUIS. A World Wide Web home page can be found at http://www.cs.cmu.edu:8001:/afs/cs.cmu.edu/project/atk-ftp/web/andrew-home.html. A book, *Multimedia Application Development with the Andrew Toolkit*, has been published by Prentice-Hall (ISBN 0-13-036633-1). An excellent tutorial is available from the Consortium by sending mail to info-andrew-request@andrew.cmu.edu and asking about the manual, *A User's Guide to AUIS*.

About the Author, Terry Gliedt

After spending over twenty years with IBM, Terry left Big Blue last year. Although he has worked with Un*x and AUIS for over six years, he is a relative new-comer to Linux. Terry does contract programming, teaches classes in C/C++ and Unix and writes the occasional technical document. You can reach him at *Software Toolsmiths*, (507) 356-4710 or by e-mail at **tpg@mr.net**.