NeXT mail vs. Unix Mail

Q: I don't get my mail on a NeXT computer, but people with NeXT computers
sometimes send me messages with NeXT Mail attachments.   How can I decipher
these using Unix mail?

A: You can decode the mail by hand if you want to.   Note that this format is likely
to change in a future release.   Follow these steps in a shell in order to read the
message:

1. At the prompt in the **mail** utility, type
          s *foo*
 to save the message as a file named   *foo.*

2. Open the file *foo* with your favorite editor.   After the mail header information is a

line   with this format:
        begin 0 *tarfile*

Remember the name that appears in place of tarfile, because uudecode will create a file with this name.

3. Decode the file   (this creates tarfile, leaving *foo* intact):
        uudecode *foo*

4. Change the permission on   tarfile and move it to the desired place.   The new filename must end with the suffix ".Z".   (Note that tarfile begins with a period so it will be invisible with a normal **ls** command; use **ls -a** instead if you want to verify that it's there.)
        chmod 666 *tarfile*
        mv *tarfile mailmsg.Z*

5. Uncompress it (this will create a file called *mailmsg* and remove *mailmsg.Z*):

uncompress *mailmsg.Z*

6.   For the final step,   *mailmsg* should be in a directory by itself so that when the message is unparsed the various files are easy to find.

     mkdir *newdir*
     mv *mailmsg newdir*
     cd *newdir*
     tar xvf *mailmsg*

The **tar** command will show you the files it has unparsed.   One of them is called index.rtf.   This is the body of the message in Rich Text Format (RTF).   The rest of the files and directories are the attachments.   Hopefully they are files that can be opened on the receiving system.   If you have an editor that can read RTF files, use it to open index.rtf .   It will have the body of the text.   There will be no text if just an attachment was sent.

The rest of this NextAnswers entry discusses what to do if your editor can't read RTF files.

Open *index.rtf* with your editor.   Much of the text in this file is RTF formatting. The curly braces and words starting with `\' are RTF directives.   You can ignore them for the most part; they introduce formatting and text style information.   The only interesting directive is   *\attachment#*.   It is the placeholder for an attachment that has been placed in the directory.   The group *{/attachment# filename }*   is the information for the attachment.

In the following example, the section in black is the attachment information.   This attachment is the file called *attachmentNam*e.   There is no regular text in this message.

{\rtf0\ansi{\fonttbl\f0\fmodern Courier;\f1\fmodern Ohlfs;}
\margl120

```
\margr120
{\attachment0 attachmentName
}
{\f1\fs20\fi0\li0\ql\gray0 \
}
}
```

In the following example, there is only text.   The words in black are the ones you are interested in:

```
{\rtf0\ansi{\fonttbl\f1\fmodern Ohlfs;\f0\fswiss Helvetica;\f3\fmodern Courier;\f2\froman Times;}
\margl120
\margr120
{\f1\fs20\fi0\li0\ql\gray0 this is some rich text}
{\f0\fs20\fi0\li0\ql\gray0  this is some helvectica\
\
this is more helvetica\
}
{\f0\fs20\fi0\li0\ql\gray0 \
}
```

```
{\f2\b\fs20\fi0\li0\ql\gray0 This is times bold}
{\f0\fs20\fi0\li0\ql\gray0 .\
\
and another line of text\
}
}
```