19. Mail Group Functions and Actions—^XMXAPIG

Errors

If any errors occur, the following variables will be defined:

XMERR The number of errors.

^TMP("XMERR",$J,<error number>,"TEXT",<line number>)=<error text>

- **$$GOTLOCAL^XMXAPIG(XMGROUP,XMDAYS)**

  This API is used to find out whether or not a mail group has any *active* local members. Messages can only be delivered to *active* local members of a mail group. Active members are described as having an Access Code and a mailbox. It sets XMERR and ^TMP("XMERR",$J), if an error occurs.

  It returns the following:

  0—No *active* members in a local mail group.

  1—Has *active* members in a local mail group.

  Usage:

```
I '$$GOTLOCAL^XMXAPIG(XMGROUP,XMDAYS) D error
```

  Input Parameters:

  XMGROUP Mail group IEN or name (exact, case-sensitive).

  XMDAYS (optional) Active members of the mail group must have used MailMan within the past number of days specified by XMDAYS. If XMDAYS is 0, null, or not supplied, it is ignored.

  Output Parameters:

  None.

  Example:

```
I '$$GOTLOCAL^XMXAPIG("GROUP") D error
```

  If the mail group named "GROUP" has no active local members, do an error routine to notify someone. Otherwise, go ahead and send the message.

Optionally, you may specify an additional constraint, that at least one member must have used MailMan in the last few days:

```
I '$$GOTLOCAL^XMXAPIG("GROUP",9) D error
```

If the mail group named "GROUP" doesn't have at least one active local member who has used MailMan in the last 9 days, do an error routine to notify someone. Otherwise, go ahead and send the message.

20. Interactive User Actions—^XM & ^XMXAPIU

^XM

These APIs may be used to creat menu options.

The primary option should be set up as follows:

Entry action: **S XMMENU(0)=<name of the menu option> D EN^XM**

Routine: xxx^XMXAPIU

Exit action: **K XMMENU D CHECKOUT^XM**

Any subordinate option should be set up as follows:

Entry action: **D CHECKIN^XM**

Routine: xxx^XMXAPIU

Exit action: **D CHECKOUT^XM**

- **EN^XM**

  Meant to be the entry action of the primary MailMan option. It sets up the user's MailMan environment and calls HEADER^XM to greet the user.

- **CHECKIN^XM**

  Meant to be the entry action of any subordinate MailMan option.

- **CHECKOUT^XM**

  Meant to be the exit action of every MailMan option.

- **HEADER^XM**

  Meant to be part of the entry action of the primary MailMan option, whether called by itself, or as part of another call, such as EN^XM. It displays a greeting to the user.

^XMXAPIU

The following are meant to be in an option's ROUTINE field. They expect that DUZ exists, and if the user is acting as a surrogate, that XMDUZ exists too. Otherwise, XMDUZ will be set to DUZ. If the XMV variables do not exist, INIT^XMVITAE will be called.

- **READ^XMXAPIU**

  Read/Manage messages in your mailbox.

   *Only the user or a surrogate may use this API.*

  Input Variables:

  XMDUZ (optional) The user whose mailbox is to be read. Default is DUZ.

  XMV (optional) The user's variables.

  Output Variables:

  None.

- **READNEW^XMXAPIU**

  Read new messages in your mailbox.

   *Only the user or a surrogate may use this API.*

  Input Variables:

  XMDUZ (optional) The user whose new messages are to be read. Default is DUZ.

  XMV (optional) The user's variables.

  Output Variables:

  None.

- **SEND^XMXAPIU**

  Send a message.

   *Only the user or a surrogate with "WRITE" privileges may use this API.*

  Input Variables:

XMDUZ	(optional) The user who is to send a message. Default is DUZ.
XMV	(optional) The user's variables.

  Output Variables:

  None.

The following is meant to be used in a routine:

- **TOWHOM^XMXAPIU(XMDUZ,XMZ,XMTYPE,.XMINSTR)**

  Ask user for message addressees.

  Input Parameters:

XMDUZ	The user (DUZ or name) who is addressing the message.
XMZ	Message number in the MESSAGE global. Not necessary, if XMTYPE="S" and XMINSTR("ADDR FLAGS") contains "R".
XMTYPE	Determines what prompts are used with the user: S—User is sending a message. F—User is forwarding a message.
.XMINSTR	(optional) Appropriate special instructions. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "ADDR FLAGS", "TO PROMPT"

  Output Parameters:

.XMINSTR	(optional) Appropriate special instructions. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "SELF BSKT", "SHARE BSKT", "SHARE DATE"

The following global variables are created:

- ^TMP("XMY0",$J) Addressee as entered by the user.
- ^TMP("XMY",$J) Resulting addressee(s) as interpreted by MailMan.

21. Message Editing—^XMXEDIT

These entry points edit different parts of a message. They do *not* perform any checks to see whether it is appropriate to do so. That is the responsibility of the calling routine.

Generally, these entry points expect that the input parameters are correct. They also expect that the calling application has assumed a level of responsibility and already taken care of the following:

- INIT^XMVVITAE Has been called to set up the user's XMV array, with vital user information, user preferences, and, if the user is a surrogate, determining level of authorization.
- Determined that the user is authorized to see the message. If the message is in the user's mailbox, then that's enough. Otherwise, $$ACCESS^XMXSEC should be used to determine authorization.
- OPTMSG^XMXSEC2 Has been called and has given its permission to edit the message or to toggle Information Only.

  

  $$EDIT^XMXSEC2 will also let you know whether the user may edit the message.
- OPTEDIT^XMXSEC2 Has been called and has given its permission to edit the particular thing we are editing here.
- INMSG2^XMXUTIL2 Has been called to set XMINSTR. These routines expect that XMINSTR has been correctly set. They will change XMINSTR according to the item being edited.

Errors

If any errors occur, the following variables may be defined:

XMERR The number of errors.

^TMP("XMERR",$J,<error number>,"TEXT",<line number>)=<error text>

- **CLOSED^XMXEDIT(XMZ,.XMINSTR,.XMMSG)**

  Toggle the message's "Closed" flag. It sets XMERR and ^TMP("XMERR",$J), if an error occurs.

  Input Parameters:

  XMZ Message IEN in the MESSAGE file (#3.9).

  .XMINSTR Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual:

  "FLAGS"

Output Parameters:

.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "FLAGS"
.XMMSG	Appropriate message, suitable for display to the user.

- **CONFID^XMXEDIT(XMZ,.XMINSTR,.XMMSG)**

  Toggle the message's "Confidential" flag. It sets XMERR and ^TMP("XMERR",$J), if an error occurs.

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "FLAGS"

  Output Parameters:

.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "FLAGS"
.XMMSG	Appropriate message, suitable for display to the user.

- **CONFIRM^XMXEDIT(XMZ,.XMINSTR,.XMMSG)**

  Toggle the message's "Confirm Receipt Requested" flag. It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "FLAGS"

Output Parameters:

.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "FLAGS"
.XMMSG	Appropriate message, suitable for display to the user.

- **DELIVER^XMXEDIT(XMZ,XMDBSKT,.XMINSTR,.XMMSG)**

  Set/Delete the message delivery basket for all users. It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
XMDBSKT	New Delivery basket name: ="**@**" (At-sign, Shift-2 key on most keyboards), if you want to delete it.

  Output Parameters:

.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: ("RCPT BSKT") Set to XMDBSKT or KILLed if XMDBSKT="**@**".
.XMMSG	Appropriate message, suitable for display to the user.

- **INFO^XMXEDIT(XMZ,.XMINSTR,.XMMSG)**

  Toggle the message's "Information Only" flag. It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "FLAGS"

Output Parameters:

.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "FLAGS"
.XMMSG	Appropriate message, suitable for display to the user.

- **PRIORITY^XMXEDIT(XMZ,.XMINSTR,.XMMSG)**

  Toggle the message's "Priority" flag. It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "FLAGS"

  Output Parameters:

.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "FLAGS"
.XMMSG	Appropriate message, suitable for display to the user.

- **SUBJ^XMXEDIT(XMZ,XMSUBJ,.XMIM)**

  Change the message subject. It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
XMSUBJ	Subject of the message. It *must* be from 3 to 65 characters in length. If null, it defaults to "*** No Subject ***". If the subject is "*** No Subject ***" and the message is sent to a remote site, the subject in the "SUBJECT:" header record will be null.

  Output Parameters:

.XMIM	Message information:	
	("SUBJ")	Message subject.

- **TEXT^XMXEDIT(XMZ,XMBODY)**

  Replace the message text. It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (3.9).
XMBODY	Closed root of array containing new message text. The root may *not* be called "XMBODY". Also, it *must* be VA FileMan WORD-PROCESSING compatible.

  Output Parameters:

  None.

- **VAPOR^XMXEDIT(XMZ,XMVAPOR,.XMINSTR,.XMMSG)**

  Set/Delete the message vaporize date. It does *not* set XMERR and ^TMP("XMERR",$J).

  This routine does not set the message vaporize date in a user's basket. Use KVAPOR^XMXUTIL to do that.

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
XMVAPOR	New message vaporize date/time. The date *must* be in VA FileMan format. ="**@**" (At-sign, Shift-2 key on most keyboards), if you want to delete it.

  Output Parameters:

.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: ("VAPOR") Set to XMVAPOR or KILLed if XMVAPOR="**@**".
.XMMSG	Appropriate message, suitable for display to the user.

22. Message Security, Permission, and Restriction Functions and Routines—^XMXSEC

Errors

If any errors occur, the following variables will be defined:

XMERR The number of errors.

^TMP("XMERR",$J,<error number>,"TEXT",<line number>)=<error text>

^XMXSEC

- **$$ACCESS^XMXSEC(XMDUZ,XMZ,XMZREC)**

  Returns a value indicating whether the user may access a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  XMDUZ User DUZ.

  XMZ Message IEN in the MESSAGE file (#3.9).

  XMZREC (optional) Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$ANSWER^XMXSEC(XMDUZ,XMZ,XMZREC)**

  Returns a value indicating whether the user may answer a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  XMDUZ User DUZ.

  XMZ Message IEN in the MESSAGE file (#3.9).

  XMZREC (optional) Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$BCAST^XMXSEC(XMZ)**

  Returns a value indicating whether a message was broadcast or not (0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

  XMZ Message IEN in the MESSAGE file (#3.9).

- **$$CLOSED^XMXSEC(XMZREC)**

  Returns a value indicating whether a message is "Closed" or not (0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Compare to $$ZCLOSED^XMXSEC described below.

  Input Parameters:

  XMZREC Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$CONFID^XMXSEC(XMZREC)**

  Returns a value indicating whether a message is "Confidential" or not (0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Compare to $$ZCONFID^XMXSEC described below.

  Input Parameters:

  XMZREC Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$CONFIRM^XMXSEC(XMZREC)**

  Returns a value indicating whether a message is "Confirm Receipt Requested" or not (0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Compare to $$ZCONFIRM^XMXSEC described below.

  Input Parameters:

  XMZREC Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$COPY^XMXSEC(XMDUZ,XMZ,XMZREC)**

  Returns a value indicating whether the user may copy a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  XMDUZ — User DUZ.

  XMZ — Message IEN in the MESSAGE file (#3.9).

  XMZREC — (optional) Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$DELETE^XMXSEC(XMDUZ,XMK,XMZ,XMZREC)**

  Returns a value indicating whether the user may delete or terminate a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  XMDUZ — User DUZ.

  XMK — Basket IEN.

  XMZ — Message IEN in the MESSAGE file (#3.9).

  XMZREC — (optional) Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$FORWARD^XMXSEC(XMDUZ,XMZ,XMZREC)**

  Returns a value indicating whether the user may forward a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  XMDUZ — User DUZ.

  XMZ — Message IEN in the MESSAGE file (#3.9).

  XMZREC — (optional) Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$INFO^XMXSEC(XMZREC)**

  Returns a value indicating whether a message is "Information Only" or not (0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Compare to $$ZINFO^XMXSEC described below.

  Input Parameters:

  XMZREC Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$LATER^XMXSEC(XMDUZ)**

  Returns a value indicating whether the user may "later" a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  XMDUZ User DUZ.

- **$$MOVE^XMXSEC(XMDUZ,XMZ,XMZREC)**

  Returns a value indicating whether the user may save or filter a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  XMDUZ User DUZ.

  XMZ Message IEN in the MESSAGE file (#3.9).

  XMZREC (optional) Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$ORIGIN8R^XMXSEC(XMDUZ,XMZREC)**

  Returns a value indicating whether the user (XMDUZ or DUZ) sent the message or not (sender or surrogate, 0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Compare to $$ZORIGIN8^XMXSEC described below.

  Input Parameters:

  XMDUZ User DUZ.

  XMZREC Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$POSTPRIV^XMXSEC( )**

  Returns a value indicating whether the user has Postmaster privileges or not, including whether or not the user may perform group message actions in SHARED,MAIL (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  None.

- **$$PRIORITY^XMXSEC(XMZREC)**

  Returns a value indicating whether a message is "Priority" or not (0=no;1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Compare to $$ZPRI^XMXSEC described below.

  Input Parameters:

XMZREC	Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$READ^XMXSEC(XMDUZ,XMZ,XMZREC)**

  Returns a value indicating whether the user may read a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

XMDUZ	User DUZ.
XMZ	Message IEN in the MESSAGE file (#3.9).
XMZREC	(optional) Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$REPLY^XMXSEC(XMDUZ,XMZ,XMZREC)**

  Returns a value indicating whether the user may reply to a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

XMDUZ	User DUZ.
XMZ	Message IEN in the MESSAGE file (#3.9).

XMZREC (optional) Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$RPRIV^XMXSEC( )**

  Returns a value indicating whether the surrogate has READ privileges or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  None.

- **$$RWPRIV^XMXSEC( )**

  Returns a value indicating whether the surrogate has READ or WRITE privileges or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  None.

- **$$SEND^XMXSEC(XMDUZ,.XMINSTR)**

  Returns a value indicating whether the user may send a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  XMDUZ User DUZ.

  .XMINSTR (optional) Appropriate special instructions. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual:

  "FROM"

- **$$SURRACC^XMXSEC(XMDUZ,XMACCESS,XMZ,XMZREC)**

  Returns a value indicating whether the surrogate may access a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  XMDUZ User DUZ.

  XMACCESS String telling type of access attempted. (Used in an error message, if access is denied.)

XMZ Message IEN in the MESSAGE file (#3.9).

XMZREC (optional) Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$SURRCONF^XMXSEC(XMDUZ,XMZ)**

  Returns a value indicating whether a message is "Confidential" or not, and if it is, whether the surrogate may access it (0=no; 1=yes, it is confidential and the surrogate may not access it). It does *not* set XMERR and ^TMP("XMERR",$J).

  This function should only be used when the user is a surrogate.

  Input Parameters:

  XMDUZ User DUZ.

  XMZ Message IEN in the MESSAGE file (#3.9).

- **$$WPRIV^XMXSEC( )**

  Returns a value indicating whether the surrogate has WRITE privileges or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  None.

- **$$ZCLOSED^XMXSEC(XMZ)**

  Returns a value indicating whether a message is "Closed" or not (0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Compare to $$CLOSED^XMXSEC described above.

  Input Parameters:

  XMZ Message IEN in the MESSAGE file (#3.9).

- **$$ZCONFID^XMXSEC(XMZ)**

  Returns a value indicating whether a message is "Confidential" or not (0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

   *Compare to $$CONFID^XMXSEC described above.*

  Input Parameters:

  XMZ Message IEN in the MESSAGE file (#3.9).

- **$$ZCONFIRM^XMXSEC(XMZ)**

  Returns a value indicating whether a message is "Confirm Receipt Requested" or not (0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

   *Compare to $$CONFIRM^XMXSEC described above.*

  Input Parameters:

  XMZ Message IEN in the MESSAGE file (#3.9).

- **$$ZINFO^XMXSEC(XMZ)**

  Returns a value indicating whether a message is "Information Only" or not (0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

   *Compare to $$INFO^XMXSEC described above.*

  Input Parameters:

  XMZ Message IEN in the MESSAGE file (#3.9).

- **$$ZORIGIN8^XMXSEC(XMDUZ,XMZ)**

  Returns a value indicating whether the user (XMDUZ or DUZ) sent the message or not (sender or surrogate, 0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Compare to $$ORIGIN8R^XMXSEC described above.

  Input Parameters:

XMDUZ	User DUZ.
XMZ	Message IEN in the MESSAGE file (#3.9).

- **$$ZPOSTPRV^XMXSEC( )**

  Returns a value indicating whether the user has Postmaster privileges or not, including whether or not the user may perform group message actions in SHARED,MAIL (0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

  None.

- **$$ZPRI^XMXSEC(XMZ)**

  Returns a value indicating whether a message is "Priority" or not (0=no;1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Compare to $$PRIORITY^XMXSEC described above.

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).

^XMXSEC1

- **$$COPYAMT^XMXSEC1(XMZ,XMWHICH)**

  This function may be used when copying a message. It checks the total number of lines and responses to be copied. Returns 1 if the amount is within site limitations; 0, if not. It sets XMERR and ^TMP("XMERR",$J), if permission is denied.

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
XMWHICH	(optional) String of which responses are to be copied. Options include: 0—Original message. Number list/range—These particular responses. Undefined/Null—Original message and all responses.

- **$$COPYLIMS^XMXSEC1( )**

  This function may be used when copying a message. Returns (three-piece ^-delimited string) the site's message copy limits. If the site has no specific limit, then MailMan defaults are used. It does *not* set XMERR and ^TMP("XMERR",$J).

  Piece 1: Number of recipients to whom the copy may be sent (default=2999).

  Piece 2: Number of responses that may be copied (default=99).

  Piece 3: Number of lines of text that may be copied (default=3999).

  Input Parameters:

  None.

- **$$COPYRECP^XMXSEC1(XMZ)**

  This function may be used when copying a message. It checks the total number of recipients on the message to see if it's "OK" to list them in the copy text and send the copy to them, too. It returns 1 if the amount is within site limitations; 0, if not. It sets XMERR and ^TMP("XMERR",$J), if permission is denied.

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).

- **$$PAKMAN^XMXSEC1(XMZ,XMZREC)**

  Returns a value indicating whether a message is a PackMan message or not (0=no; 1=yes). It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

  XMZ — Message IEN in the MESSAGE file (#3.9).

  XMZREC — (optional) Zero node of the message: ^XMB(3.9,XMZ,0).

- **CHKLINES^XMXSEC1(XMDUZ,XMZ,.XMRESTR)**

  Checks whether a message is too long to be sent to a remote site. If $D(XMRESTR("NONET")), then it is. It does *not* set XMERR and ^TMP("XMERR",$J).

  This routine does not KILL XMRESTR.

  Input Parameters:

  XMDUZ — User DUZ.

  XMZ — Message IEN in the MESSAGE file (#3.9).

  Output Parameters:

  If the message is not too long or if the user holds the XMMGR security key, then XMRESTR("NONET") is *not* set. Otherwise, XMRESTR("NONET") equals the maximum number of lines a message may have when sending to a remote site.

- **CHKMSG^XMXSEC1(XMDUZ,XMK,XMKZ,XMZ,XMZREC)**

  Checks whether or not the message is located where the calling routine says it is, and whether or not the user may access it. It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

  XMDUZ — DUZ of the user who is accessing the message.

  XMK, XMKZ — Message being accessed. For a description of these parameters, please refer to the "Parameter Definitions" list in Chapter 16 in this manual.

Output Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
XMZREC	Zero node of the message: ^XMB(3.9,XMZ,0).

- **GETRESTR^XMXSEC1(XMDUZ,XMZ,XMZREC,.XMINSTR,.XMRESTR)**

  Returns assorted restrictions, if any, on sending or forwarding the message. It does *not* set XMERR and ^TMP("XMERR",$J).

   *This routine does not KILL XMRESTR.*

  Input Parameters:

XMDUZ	User DUZ.
XMZ	Message IEN in the MESSAGE file (#3.9).
XMZREC	(optional) Zero node of the message: ^XMB(3.9,XMZ,0).
.XMINSTR	(optional) Appropriate special instructions. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "ADDR FLAGS"

  Output Parameters:

.XMRESTR	Restrictions on forwarding the message. Here are the nodes that may be set: ➢ If $G(XMRESTR("NONET")), then the message is too long to be sent to a remote site, and it equals the maximum number of lines a message may have when sending to a remote site. ➢ If $G(XMRESTR("FLAGS"))["C", then the message may *not* be forwarded to SHARED,MAIL (because it is confidential). ➢ If $G(XMRESTR("FLAGS"))["X", then the message may *not* be forwarded to SHARED,MAIL (because it is closed). ➢ If $D(XMRESTR("NOFPG")), then the message may *not* be forwarded to groups (because it is a priority message, the user didn't send it, the user doesn't possess the XM GROUP PRIORITY security key, and XMINSTR("ADDR FLAGS")'["R").

- **OPTGRP^XMXSEC1(XMDUZ,XMK,.XMOPT)**

  Determines what the user may do at the basket or message group level. It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

  XMDUZ — DUZ of the user who is accessing the basket.

  XMK — Basket IEN.

  Output Parameters:

  .XMOPT — Commands which the user may use are a subset of the following:

  If the user may not use them, they either won't appear (only the Postmaster will see "**X**") or they will have subnode(s) under "**?**" with an explanation as to why they may *not* be used.

  For example, if the basket were the "IN" basket, then XMOPT("C","?")="The name of the IN basket may not be changed."

  If there were **n** lines of text, then XMOPT("C","?",1), ... XMOPT("C","?",**n**-1), and XMOPT("C","?") would be set.

  ("C") "Change the name of this basket"

  ("D") "Delete messages"

  ("F") "Forward messages

  ("FI") "Filter messages"

  ("H") "Headerless Print messages"

  ("L") "Later messages"

  ("N") "New messages list"

  ("P") "Print messages"

  ("Q") "Query (search for) messages"

  ("R") "Resequence messages"

  ("S") "Save messages"

  ("T") "Terminate messages"

  ("X") "Xmit priority toggle"

^XMXSEC2

- **$$EDIT^XMXSEC2(XMDUZ,XMZ,XMZREC)**

  Returns a value indicating whether the user may edit a message or not (0=no; 1=yes). It sets XMERR and ^TMP("XMERR",$J), if access/permission is denied.

  Input Parameters:

XMDUZ	User DUZ.
XMZ	Message IEN in the MESSAGE file (#3.9).
XMZREC	(optional) Zero node of the message: ^XMB(3.9,XMZ,0).

- **OPTEDIT^XMXSEC2(.XMINSTR,.XMOPT)**

  If OPTMSG^XMXSEC2 determines that the user may edit the message, then OPTEDIT determines what, exactly, the user may edit. It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

.XMINSTR	Set by INMSG2^XMXUTIL2: "FLAGS", "TYPE", "VAPOR", "RCPT BSKT", "SCR HINT"

  Output Parameters:

.XMOPT	Commands which the user may use are a subset of the following: If the user may not use them, they will have subnode(s) under "**?**" with an explanation as to why they may not be used. ("C") "Confidential set/remove" ("D") "Delivery basket set/remove" ("P") "Priority/Normal message" ("R") "Confirm receipt set/remove" ("S") "edit Subject" ("T") "edit Text" ("V") "Vaporize date set/remove" ("X") "Closed message set/remove"

- **OPTMSG^XMXSEC2(XMDUZ,XMK,XMZ,.XMIM,.XMINSTR,.XMIU,.XMOPT)**

  Determines what the user may do with the message. Some input parameters are set by calling INMSG1 and INMSG2^XMXUTIL2. It does *not* set XMERR and ^TMP("XMERR",$J).

  Input Parameters:

  XMDUZ — DUZ of the user who is accessing the message.

  XMK — Basket IEN where the message is located:

  0—if not in basket.

  XMZ — Message IEN in the MESSAGE file (#3.9).

  .XMIM — Message information, set by INMSG1^XMXUTIL2:

  ("FROM") — Who sent the message.

  .XMINSTR — Set by INMSG2^XMXUTIL2

  "FLAGS", "TYPE", "VAPOR", "RCPT BSKT", "SCR HINT"

  .XMIU — User information, as related to the message:

  ("ORIGN8") — Did the user send the message? Set by INMSG2^XMXUTIL2.

  ("IEN") — User IEN in message RECIPIENT multiple, set by INMSG1^XMXUTIL2.

  Output Parameters:

  .XMOPT — Commands which the user may use are a subset of the following:

  If the user may not use them, they will have subnode(s) under "**?**" with an explanation as to why they may not be used.

  For example, if the message is Information Only, then XMOPT("R","?")="Only the sender may Reply to an 'Information only' message."

  If there were **n** lines of text, then XMOPT("R","?",1), ... XMOPT("R","?",**n**-1), and XMOPT("R","?") would be set.

  ("A") "Answer"

  ("AA") "Access Attachments"

  ("B") "Backup"

  ("C") "Copy"

  ("D") "Delete"

  ("E") "Edit"

  ("F") "Forward"

  ("I") "Ignore"

  ("IN") "Information Only set/remove"

("H") "Headerless Print"
("K") "Priority replies set/remove"
("L") "Later"
("N") "New"
("P") "Print"
("Q") "Query"
("QR") "Query Recipients"
("QD") "Query Detailed"
("QN") "Query Network"
("R") "Reply"
("S") "Save"
("T") "Terminate"
("V") "Vaporize date edit"
("W") "Write"
("X") "Xtract KIDS/PackMan"

23. Message and Mailbox Utility Functions and Routines—^XMXUTIL

- **$$BMSGCT^XMXUTIL(XMDUZ,XMK)**

  Returns the number of messages in a user's basket.

  Input Parameters:

  XMDUZ User DUZ.

  XMK Basket IEN.

- **$$BNMSGCT^XMXUTIL(XMDUZ,XMK)**

  Returns the number of new messages in a user's basket.

  Input Parameters:

  XMDUZ User DUZ.

  XMK Basket IEN.

- **$$BSKTNAME^XMXUTIL(XMDUZ,XMK)**

  Returns the name of a user's basket.

  Input Parameters:

  XMDUZ User DUZ.

  XMK Basket IEN.

- **$$NAME^XMXUTIL(XMDUZ,XMINFO)**

  Returns the name of the user by looking up XMDUZ in the NEW PERSON file (#200). Optionally, it may also return the user's Title and/or Institution. If XMDUZ is not numeric, it returns XMDUZ.

  Input Parameters:

  XMDUZ — User DUZ.

  XMINFO — (optional) If the variables XMV("SHOW INST") and XMV("SHOW TITL") indicate that the user's institution and/or title are desired, should that information be returned, too?

  0—no (default)

  1—yes

- **$$NETNAME^XMXUTIL(XMDUZ)**

  Returns network name of user, including @site name.

  Input Parameters:

  XMDUZ — User DUZ or any string.

- **$$NEWS^XMXUTIL(XMDUZ,XMTEST)**

  Returns information on new messages in a user's mailbox. This function returns much the same information as the routine QMBOX^XMXAPI. It returns the following:

  -1—If XMDUZ is not a valid user.

  0—If the user has no new messages.

  Otherwise, it returns the following ^-delimited string:

  Piece 1: Number of new messages in the mailbox.

  Piece 2: Does the user have new priority mail?

  0—no

  1—yes

  Piece 3: Number of new messages in the "IN" basket.

  Piece 4: Date/time (in VA FileMan format) that the last message was received.

  Piece 5: Have there been any new messages since the last time this routine was called?

  0—no

  1—yes

Input Parameters:

XMDUZ	User DUZ.
XMTEST	(optional) Is this a test? 0—no 1—yes (default) If this is *not* a test, then the LAST NEW MSG NOTIFY DATE/TIME field (#1.12) in the MAILBOX file (#3.7) may be updated for this user.

- **$$TMSGCT^XMXUTIL(XMDUZ)**

  Returns the total number of messages in a user's mailbox.

  Input Parameters:

XMDUZ	User DUZ.

- **$$TNMSGCT^XMXUTIL(XMDUZ)**

  Returns the total number of new messages in a user's mailbox.

  Input Parameters:

XMDUZ	User DUZ.

- **KVAPOR^XMXUTIL(XMDUZ,XMK,XMZ,XMVAPOR,.XMIU)**

  Sets/Removes a message vaporize date in a user's basket.

  Input Parameters:

XMDUZ	User DUZ.
XMK	Basket IEN.
XMZ	Message IEN in the MESSAGE file (#3.9).
XMVAPOR	Date/time (in VA FileMan format) to delete this message from this user's basket: ="**@**" (At-sign, Shift-2 key on most keyboards) to remove the vaporize date.

  Output Parameters:

.XMIU	User information, as related to the message: ("KVAPOR") Set to XMVAPOR or KILLed if XMVAPOR="**@**"

- **LASTACC^XMXUTIL(XMDUZ,XMK,XMZ,XMRESP,.XMIM,.XMINSTR,.XMIU, .XMCONFRM)**

  Record that the user has read the message. This routine needs to be called only by those applications that display, using their own routines, messages and responses to the user. This routine sets the first and last times that the user has read the message. It records the last response that the user has read. If the user is a surrogate, it records the surrogate was the last reader. If MailMan had set a vaporize date for the message in the user's basket (because the user hadn't accessed it in a while), then that vaporize date is deleted. It also sends a confirmation message to the sender, if one was requested, the first time the user reads the message.

  Input Parameters:

XMDUZ	User DUZ.	
XMK	Basket IEN.	
XMZ	Message IEN in the MESSAGE file (#3.9).	
XMRESP	Last response read by the user this time.	
.XMIM	Message information, set by INMSG1^XMXUTIL2:	
	("SUBJ")	Subject.
	("FROM")	Sender.
.XMINSTR	More message information, set by INMSG2^XMXUTIL2:	
	("FLAGS")	Special instructions (here, we are interested in whether "FLAGS"["R"—confirm receipt requested).
.XMIU	User information, as related to the message:	
	("IEN")	IEN of user record in message RECIPIENT multiple, set by INMSG1^XMXUTIL2.
	("RESP")	Last response read by the user, initially set by INMSG1^XMXUTIL2 or INRESPS^XMXUTIL2.

  Output Parameters:

.XMIU	User information, as related to the message:	
	("RESP")	If XMRESP is greater than XMIU("RESP"), then XMIU("RESP") is set to XMRESP.
.XMCONFRM	Was a confirmation message sent to the message sender?	
	0—no	
	1—yes	

- **MAKENEW^XMXUTIL(XMDUZ,XMK,XMZ,XMLOCKIT)**

  Makes a message new and updates the new message counts.

  Input Parameters:

XMDUZ	User DUZ.
XMK	Basket IEN.
XMZ	Message IEN in the MESSAGE file (#3.9).
XMLOCKIT	(optional) Should MailMan take care of locking and unlocking the ^XMB(3.7,XMDUZ global? 0—no (default) 1—yes

  
  The locking must be done to ensure the integrity of the new message counts. If MailMan doesn't do it, then the calling application must.

  Output Parameters:

  None.

- **NONEW^XMXUTIL(XMDUZ,XMK,XMZ,XMLOCKIT)**

  Makes a message *not* new and updates the new message counts.

  Input Parameters:

XMDUZ	User DUZ.
XMK	Basket IEN.
XMZ	Message IEN in the MESSAGE file (#3.9).
XMLOCKIT	(optional) Should MailMan take care of locking and unlocking the ^XMB(3.7,XMDUZ global? 0—no (default) 1—yes

  
  The locking must be done to ensure the integrity of the new message counts. If MailMan doesn't do it, then the calling application must.

Output Parameters:

None.

- **PAGE^XMXUTIL(.XMABORT)**

  Displays to the user: "Enter RETURN to continue or '^' to exit:" and waits until the user presses a key. Sets up and uses the standard VA FileMan call to do this.

  Input Parameters:

  None.

  Output Parameters:

  .XMABORT Did the user choose to exit? (0=no; 1=yes)

- **WAIT^XMXUTIL**

  Displays to the user: "Press RETURN to continue:" and waits until the user presses a key. Sets up and uses the standard VA FileMan call to do this.

  Input Parameters:

  None.

  Output Parameters:

  None.

24. Date and String Utility Functions and Routines—^XMXUTIL1

- **$$CONVERT^XMXUTIL1(X,XMTIME)**

  Given an Internet DATE/TIME string, it returns the VA FileMan DATE/TIME. If the Internet DATE/TIME string *cannot* be understood, it returns -1.

  Input Parameters:

X	Internet DATE/TIME.
XMTIME	(optional) Should the time also be converted? 0—no (default) 1—yes If no, it returns the VA FileMan date only.

- **$$CTRL^XMXUTIL1(XMSTRING)**

  Strip control characters from a string.

  Input Parameters:

XMSTRING	The string.

- **$$DECODEUP^XMXUTIL1(XMSTRING)**

  Change all **~U~** to ^ in a string.

  Input Parameters:

XMSTRING	The string.

- **$$ENCODEUP^XMXUTIL1(XMSTRING)**

  Change all ^ to **~U~** in a string.

  Input Parameters:

XMSTRING	The string.

- **$$GMTDIFF^XMXUTIL1(XMZONE)**

  Given the time zone, it returns **+-hhmm** difference from Greenwich Mean Time (GMT). If there's no record of the time zone, it returns the null string.

  Input Parameters:

  XMZONE The 3-character time zone.

- **$$INDT^XMXUTIL1(XMDT)**

  Given the VA FileMan DATE/TIME, it returns the Internet DATE/TIME string:

  dd mm yyy hh:mm:ss +-hhmm (time zone)

  Input Parameters:

  XMDT VA FileMan DATE/TIME.

- **$$MAXBLANK^XMXUTIL1(XMSTRING)**

  Reduce all three or more consecutive blanks in a string to two.

  Input Parameters:

  XMSTRING The string.

- **$$MELD^XMXUTIL1(XMSTRING,XMNUMBER,XMLEN)**

  Combine a string and a number to form a new string of a given length. The string will be right justified; the number left-justified, with at least two blanks separating the string and number. The string will be truncated, if necessary.

  For example:

  $$MELD^XMXUTIL1("Lotus blossom",123,10) returns "Lotus 123"

  Input Parameters:

  XMSTRING The string.

  XMNUMBER (optional) The number.

  XMLEN The length of the new string to be formed.

- **$$MMDT^XMXUTIL1(XMDT)**

  Given the VA FileMan DATE/TIME, it returns the following string:

  dd mmm yy hh:mm

  Input Parameters:

  XMDT VA FileMan DATE/TIME.

- **$$SCRUB^XMXUTIL1(XMSTRING)**

  Strip control characters and leading/trailing blanks from a string.

  Input Parameters:

  XMSTRING The string.

- **$$STRIP^XMXUTIL1(XMSTRING)**

  Strip leading/trailing blanks from a string.

  Input Parameters:

  XMSTRING The string.

- **$$TIMEDIFF^XMXUTIL1(XMDIFF)**

  Given the decimal time difference (between time zones), it returns **+-hhmm** (e.g., **-2.5** equates to **–0230**).

  Input Parameters:

  XMDIFF Decimal time difference.

- **$$TSTAMP^XMXUTIL1( )**

  Return a timestamp (**$H** expressed in seconds).

  Input Parameters:

  None.

- **ZONEDIFF^XMXUTIL1(XMYT,.XMHH,.XMMM)**

  Given the time zone (or time difference **+-hhmm** from Greenwich Mean Time [GMT]), it returns the number of hours and minutes difference between that and the local time zone.

  Input Parameters:

  XMYT Time zone (3-character or **+-hhmm** from GMT).

  Output Parameters:

  .XMHH Number of hours time difference.

  .XMMM Additional number of minutes time difference.

25. Message Information Functions and Routines—^XMXUTIL2 & ^XMXUTIL3

These functions and routines retrieve all kinds of information about a message:

- Information that can be displayed.
- Information that can be used to determine what may (and may not) be done with the message.

^XMXUTIL2

- **$$BSKT^XMXUTIL2(XMDUZ,XMZ,XMNAME)**

  Returns which basket a message is in for a user. It returns the following:

  0—Not in a basket for this user.

  Number—It's in this basket IEN for the user. (XMNAME=0)

  Number^name—It's in this basket IEN of this name for the user. (XMNAME=1)

  Input Parameters:

XMDUZ	User DUZ.
XMZ	Message IEN in the MESSAGE file (#3.9).
XMNAME	(optional) Return the basket name, too? 0—no (default) 1—yes

- **$$DATE^XMXUTIL2(XMZREC,XMTIME)**

  Returns the message sent date. It is returned in external format:

  DD MMM YY HH:MM

   *Compare to $$ZDATE^XMXUTIL2 described below.*

  Input Parameters:

XMZREC	Zero node of the message: ^XMB(3.9,XMZ,0).
XMTIME	(optional) Return the time, also? 0—no, date only 1—yes, date and time (default)

- **$$FROM^XMXUTIL2(XMZREC)**

  Returns the message From information. It is returned in external format.

   *Compare to $$ZFROM^XMXUTIL2 described below.*

  Input Parameters:

  XMZREC Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$KSEQN^XMXUTIL2(XMDUZ,XMK,XMZ)**

  Returns the sequence number for a message in this user's basket.

  Input Parameters:

  XMDUZ User DUZ.

  XMK Basket IEN.

  XMZ Message IEN in the MESSAGE file (#3.9).

- **$$LINE^XMXUTIL2(XMZ)**

  Returns the number of lines in the text of a message.

  Input Parameters:

  XMZ Message IEN in the MESSAGE file (#3.9).

- **$$NEW^XMXUTIL2(XMDUZ,XMK,XMZ)**

  Returns a value indicating whether or not a message is new for this user in this basket (0=no;1=yes).

  Input Parameters:

  XMDUZ User DUZ.

  XMK Basket IEN.

  XMZ Message IEN in the MESSAGE file (#3.9).

- **$$PRI^XMXUTIL2(XMZREC)**

  Returns a value indicating whether the message is priority or not (0=no;1=yes).

   *Compare to $$ZPRI^XMXUTIL2 described below.*

  Input Parameters:

  XMZREC — Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$QRESP^XMXUTIL2(XMZ,XMZREC,XMWHICH)**

  Determines whether a message is a response or not. It returns the following:

  0—Message XMZ is not a response.

  IEN^number—It's a response to this message IEN in the MESSAGE file (#3.9). And it is response **<number>** to that message. This 2nd piece is only returned if XMWHICH=1.

  Input Parameters:

  XMZ — Message IEN in the MESSAGE file (#3.9).

  XMZREC — (optional) Zero node of the message: ^XMB(3.9,XMZ,0).

  XMWHICH — (optional) If it is a response to a message, do you want to know which number response?

  0—no (default)

  1—yes

- **$$RESP^XMXUTIL2(XMZ)**

  Returns the number of responses to a message.

  Input Parameters:

  XMZ — Message IEN in the MESSAGE file (#3.9).

- **$$SUBJ^XMXUTIL2(XMZREC)**

  Returns the message subject. It is returned in external format.

   *Compare to $$ZSUBJ^XMXUTIL2 described below.*

  Input Parameters:

XMZREC	Zero node of the message: ^XMB(3.9,XMZ,0).

- **$$ZDATE^XMXUTIL2(XMZ,XMTIME)**

  Returns the message sent date. It is returned in external format:

  DD MMM YY HH:MM.

   *Compare to $$DATE^XMXUTIL2 described above.*

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
XMTIME	(optional) Return the time, also? 0—no, date only 1—yes, date and time (default)

- **$$ZFROM^XMXUTIL2(XMZ)**

  Returns the message From. It is returned in external format.

   *Compare to $$FROM^XMXUTIL2 described above.*

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).

- **$$ZNODE^XMXUTIL2(XMZ)**

  Returns the message zero node: ^XMB(3.9,XMZ,0).
  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).

- **$$ZPRI^XMXUTIL2(XMZ)**

  Returns a value indicating whether the message is priority or not (0=no;1=yes).

   *Compare to $$PRI^XMXUTIL2 described above.*

  Input Parameters:

  XMZ Message IEN in the MESSAGE file (#3.9).

- **$$ZREAD^XMXUTIL2(XMDUZ,XMZ)**

  Returns the number of responses to a message this user has read. It returns the following:

  null—User has not read the message at all.

  0—User has read the original message only.

  Number—User has read through this response.

  Input Parameters:

  XMDUZ User DUZ.

  XMZ Message IEN in the MESSAGE file (#3.9).

- **$$ZSUBJ^XMXUTIL2(XMZ)**

  Returns the message subject. It is returned in external format.

   *Compare to $$SUBJ^XMXUTIL2 described above.*

  Input Parameters:

  XMZ Message IEN in the MESSAGE file (#3.9).

- **INMSG^XMXUTIL2(XMDUZ,XMK,XMZ,XMZREC,XMFLAGS,.XMIM,.XMINSTR, .XMIU)**

  Message information.

  

  This routine should only be called for messages, not for responses. This routine calls both INMSG1^XMXUTIL2 and INMSG2^XMXUTIL2. It also returns additional information.

  Input Parameters:

XMDUZ	User DUZ.
XMK	Basket IEN. (Set XMK=0, if the message is *not* in a basket or if you are *not* interested in variables XMIU("KVAPOR") and XMIU("NEW").)
XMZ	Message IEN in the MESSAGE file (#3.9).
XMZREC	(optional) Zero node of the message: ^XMB(3.9,XMZ,0).
XMFLAGS	(optional) Used to control output: I—Internal values only. (Default is internal values, and, where it makes sense, to set variables with other values, too.) F—Set variable with internal VA FileMan date format. (Default is external MailMan date format.) "F" is ignored if XMFLAGS contains "I".

  Output Parameters:

.XMIM	Message information (KILLed first). For a description of this parameter, please refer to the definition of XMIM for INMSG1^XMXUTIL2 (described below): "SUBJ", "ENV FROM", "FROM", "FROM DUZ", "FROM NAME", "DATE", "DATE FM", "DATE MM", "SENDR", "SENDR DUZ", "SENDR NAME", "LINES", "RESPS", "RECIPS", "CRE8", "CRE8 MM"
.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "FLAGS", "TYPE", "VAPOR", "RCPT BSKT", "SCR HINT"

.XMIU — User information, as related to the message. For a description of this parameter, please refer to the definition of XMIU for INMSG1^XMXUTIL2 and INMSG2^XMXUTIL2 (described below):

"IEN", "RESP", "ORIGN8"

("KVAPOR") — DATE/TIME (in VA FileMan format) to delete this message from this user's basket. (Set only if applicable.)

("NEW") — Is message new?

0—no

1—yes

2—yes, and priority, too.

The following table compares the variables returned by $$HDR^XMGAPI2 and INMSG^XMXUTIL2:

$$HDR^XMGAPI2	INMSG^XMXUTIL2
L("BLOBCNT")	N/A
L("BROADCAST")	N/A (use $$BCAST^XMXSEC)
L("BSKT IEN")	N/A (use $$BSKT^XMXUTIL2)
L("BSKT")	N/A (use $$BSKT^XMXUTIL2)
L("DATE FM")	XMIM("DATE FM")—If XMFLAGS["F" and XMFLAGS'["I".
L("DATE")	XMIM("DATE")—Internal.
L("LINES")	XMIM("LINES")
L("NEW")	XMIU("NEW")
L("PXMZ")	N/A
L("RRCV")	XMIM("RESPS")
L("RRED")	XMIU("RESP")
L("RSP",i)	N/A (use INRESP^XMXUTIL2 to get response information)
L("SENDER DUZ")	XMIM("FROM DUZ")—If XMFLAGS'["I".
L("SENDER")	XMIM("FROM NAME")—If XMFLAGS'["I".
L("SUBJ")	XMIM("SUBJ")
L("SURROG")	XMIM("SENDR NAME")—If XMFLAGS'["I".
L("TYPE")	XMINSTR("TYPE")
L("XMZ")	XMIM("XMZ")
N/A	XMIM("CRE8")—Local create date.
N/A	XMIM("CRE8 MM")—MailMan external formatted date, if XMFLAGS'["F" or "I".

Table 4: Comparison of Variables Returned By $$HDR^XMGAPI2 and INMSG^XMXUTIL2

Table 4 (continued):

$$HDR^XMGAPI2	INMSG^XMXUTIL2
N/A	XMIM("DATE MM")—MailMan external formatted date, if XMFLAGS'["F" or "I".
N/A	XMIM("ENV FROM")—Message envelope "MAIL FROM:"
N/A	XMIM("FROM")—Internal.
N/A	XMIM("RECIPS")—Number of recipients.
N/A	XMIM("SENDR DUZ")—If XMFLAGS'["I".
N/A	XMIM("SENDR")—Internal.
N/A	XMINSTR("FLAGS")—Closed, Confidential, Information Only, Priority, Confirmation requested, Priority responses.
N/A	XMINSTR("RCPT BSKT")—Delivery basket.
N/A	XMINSTR("SCR HINT")—Scramble hint.
N/A	XMINSTR("VAPOR")—Vaporize date of message.
N/A	XMIU("IEN")—User's IEN in RECIPIENT multiple.
N/A	XMIU("KVAPOR")—Vaporize date of message in user's basket.
N/A	XMIU("ORIGN8")—Did user send message?

Table 4**: Comparison of Variables Returned By $$HDR^XMGAPI2 and INMSG^XMXUTIL2 (continued)**

- **INMSG1^XMXUTIL2(XMDUZ,XMZ,XMZREC,XMFLAGS,.XMIM,.XMIU)**

  Message information, Part 1.

  This routine should only be called for messages, not for responses. It calls routine INRESPS^XMXUTIL2.

  Input Parameters:

XMDUZ	User DUZ.
XMZ	Message IEN in the MESSAGE file (#3.9).
XMZREC	(optional) Zero node of the message: ^XMB(3.9,XMZ,0).
XMFLAGS	(optional) Used to control setting of output variables: I—Internal values only (Default is internal values, and, where it makes sense, to set variables with other values, too.) F—Set variable with internal VA FileMan date format. (Default is external MailMan date format.) "F" is ignored if XMFLAGS contains "I".

  Output Parameters:

.XMIM	Message information (KILLed first):	
	("XMZ")	Message IEN in the MESSAGE file (#3.9).
	("SUBJ")	Subject of message (all **~U~** translated to ^).
	("ENV FROM")	"MAIL FROM:", as stated in the message envelope on a message that was received from a remote site. (*Not* set, if it doesn't exist.)
	("FROM")	Who sent the message (internal).
	("FROM DUZ")	DUZ of person who sent the message (if applicable). (*Not* set if XMFLAGS contains "I".)
	("FROM NAME")	Name of person who sent the message. (*Not* set if XMFLAGS contains "I".)
	("DATE")	When the message was sent (internal).
	("DATE FM")	VA FileMan date (-1, if error). (Set if XMFLAGS contains "F". *Not* set if XMFLAGS contains "I".)

	("DATE MM")	External MailMan format: **dd mmm yy hh:mm** (Internet date, if error.) (*Not* set, if XMFLAGS contains "I" or "F".)
	("CRE8")	Local create date (in VA FileMan format).
	("CRE8 MM")	External MailMan format: **dd mmm yy hh:mm** (*Not* set, if XMFLAGS contains "I" or "F".)
	("SENDR")	Who really sent the message (if applicable).
	("SENDR DUZ")	DUZ of person who really sent the message (if applicable). (*Not* set, if XMFLAGS contains "I".)
	("SENDR NAME")	Name of person who really sent the message (if applicable). (*Not* set, if XMFLAGS contains "I".)
	("LINES")	How many lines are in the message.
	("RESPS")	How many responses does the message have.
.XMIU	User information, as related to the message (KILLed first).	
	("IEN")	IEN of XMDUZ in the message's RECIPIENT multiple.
	("RESP")	Number of the last response that the user has read.

- **INMSG2^XMXUTIL2(XMDUZ,XMZ,XMZREC,.XMIM,.XMINSTR,.XMIU)**

  Message information, Part 2.

  This routine should only be called for messages, not for responses.

  Input Parameters:

XMDUZ	User DUZ.
XMZ	Message IEN in the MESSAGE file (#3.9).
XMZREC	(optional) Zero node of the message: ^XMB(3.9,XMZ,0).

Output Parameters:

.XMIM	Message information. ("RECIPS") Number of recipients of the message.
.XMINSTR	Special instructions on the message. For a description of this parameter, please refer to the "Parameter Definitions" list in Chapter 16 in this manual: "FLAGS", "TYPE", "VAPOR", "RCPT BSKT", "SCR HINT"
.XMIU	User information, as related to the message: ("ORIGN8") Did the user send the message? 0—no 1—yes (results from a call to $$ORIGIN8R^XMXSEC)

- **INRESP^XMXUTIL2(XMZ,XMWHICH,XMFLAGS,.XMIR)**

  Response information.

  This routine should only be called for responses, not for messages.

  Input Parameters:

XMDUZ	User DUZ.
XMZ	Message IEN in the MESSAGE file (#3.9).
XMWHICH	The number of the response for which to get the information.
XMFLAGS	(optional) Used to control output: I—Internal values only. (Default is internal values, and, where it makes sense, to set variables with other values, too.) F—Set variable with internal VA FileMan date format. (Default is external MailMan date format.) "F" is ignored if XMFLAGS contains "I".

  Output Parameters:

.XMIR	Response information (KILLed first). For a description of this parameter, please refer to the definition of XMIM for INMSG1^XMXUTIL2 (previously described): "XMZ", "SUBJ", "ENV FROM", "FROM", "FROM DUZ", "FROM NAME", "DATE", "DATE FM", "DATE MM", "SENDR", "SENDR DUZ", "SENDR NAME", "LINES"

- **INRESPS^XMXUTIL2(XMZ,.XMIM,.XMIU)**

  How many responses? What has the user read?

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
.XMIU("IEN")	Set by INMSG1^XMXUTIL2.

  Output Parameters:

.XMIM("RESPS")	Number of responses for a message.
.XMIU("RESP")	Number of the last response that the user has read.

^XMXUTIL3

- **Q^XMXUTIL3(XMZ,XMFLAGS,XMAMT,.XMSTART,XMFIND,XMTROOT)**

  Get a list of the addressees of this message. Optionally, find addressees, which match a string. Gets a list (similar in format to that produced by LIST^DIC) of the requested addressees.

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
XMFLAGS	(Reserved for future use.)
XMAMT	(optional) How many? Number—Get this many. *****—Get all (default).
.XMSTART	(optional) Used to start the Lister. The Lister keeps it updated from call to call. ("IEN") Start *after* this addressee IEN. Continues from there, with each successive call, to the end. (Default is to start at the beginning.)
XMFIND	(optional) Find addressees that match the string. (VA FileMan's FIND^DIC is used.) If XMFIND is supplied, then XMAMT and XMSTART are ignored; a complete list is always returned.
XMTROOT	Target root (closed) to receive the message list. (Default is ^TMP("XMLIST",$J).)

  Output Parameters:

.XMSTART	(As defined above.)
XMTROOT	Fields returned under XMTROOT for each addressee: "TO NAME" Addressee name. "TYPE" Addressee type (if present): I—Information Only C—cc (Carbon Copy)

- **QD^XMXUTIL3(XMZ,XMFLAGS,XMAMT,.XMSTART,XMFIND,XMTROOT)**

  Get a list of the recipients of this message. Optionally, find recipients that match a string. Gets a list (similar in format to that produced by LIST^DIC) of the requested recipients.

  Input Parameters:

  XMZ — Message IEN in the MESSAGE file (#3.9).

  XMFLAGS — (Reserved for future use.)

  XMAMT — (optional) How many?

  Number—Get this many

  *****—Get all (default)

  .XMSTART — (optional) Used to start the Lister going. The Lister will keep it updated from call to call.

  ("IEN") — Start *after* this recipient IEN.

  Continues from there, with each successive call, to the end. (Default is to start at the beginning.)

  XMFIND — (optional) Find recipients that match the string. (VA FileMan's FIND^DIC is used.) If XMFIND is supplied, then XMAMT and XMSTART are ignored; a complete list is always returned.

  XMTROOT — Target root (closed) to receive the message list. (Default is ^TMP("XMLIST",$J).)

  Output Parameters:

  .XMSTART — (As defined above.)

  XMTROOT — Fields returned under XMTROOT for each recipient:

  "TO" — Recipient .01 field (could be DUZ or text).

  "TO NAME" — Recipient name.

  "TO ID" — ID of recipient:

  L—Local user

  F—Fax

  R—Remote

  S—Server

  D—Device

  *****—Broadcast

"TYPE"	(if present) Recipient type: I—Information Only C—cc (Carbon Copy)
"FWD BY DUZ"	(if present) DUZ of the person who forwarded to this recipient.
"FWD BY"	(if present) Name of the person, possibly followed by, in parentheses, the name of the surrogate of the person, who forwarded to this recipient.
"FWD ON"	Date that message was forwarded to this recipient in MailMan format: **dd mmm yy hh:mm**

Depending on "TO ID", the following fields are also returned:

- **"TO ID"="L"—Local User:**

"TO DUZ"	DUZ of the local recipient.
"RESP"	(if present) Number of the last response read (zero equals original message).
"LREAD"	(if present) DATE/TIME (in VA FileMan format) the message was last read.
"LREAD MM"	(if present) DATE/TIME (in MailMan format) the message was last read.
"FREAD"	(if present) DATE/TIME (in VA FileMan format) the message was first read.
"FREAD MM"	(if present) DATE/TIME (in MailMan format) the message was first read.
"COPY"	(if present) DATE/TIME (in VA FileMan format) the message was last copied.
"COPY MM"	(if present) DATE/TIME (in MailMan format) the message was last copied.
"TERM"	(if present) DATE/TIME (in VA FileMan format) the message was terminated.
"TERM MM"	(if present) DATE/TIME (in MailMan format) the message was terminated.
"SURR"	(if present) Name of the surrogate who last read the message.

- **"TO ID"="*"—Broadcast:**

  No additional fields.

- **"TO ID"="F"—Fax:**

"FDATE"	(if present) DATE/TIME (in VA FileMan format) the message was passed to the Fax software.
"FDATE MM"	(if present) DATE/TIME (in MailMan format) the message was passed to the Fax software.
"STATUS"	(if present) Status of the fax (present before the message is passed to the Fax software).
"FAX IEN"	(if present) IEN (in FAX ROLODEX file) of the fax recipient (present before the message is passed to the Fax software).
"ID"	(if present) Fax ID (present after the message is passed to the Fax software).

- **"TO ID"="R"—Remote:**

"XDATE"	(if present) DATE/TIME (in VA FileMan format) the transmission of the message began (present after transmission is complete).
"XDATE MM"	(if present) DATE/TIME (in MailMan format) the transmission of the message began (present after transmission is complete).
"STATUS"	(if present) Status of the message (present before and during transmission, and if error, afterward).
"ID"	(if present) Message ID (present after transmission is successfully completed).
"PATH"	(if present) IEN (in DOMAIN file [#4.2]) of the Domain to/via which the message will be sent (present before and during transmission).
"PATH NAME"	(if present) Name of the Domain to/via which the message will be sent (present before and during transmission).
"SECS"	(if present) Duration of the transmission (in seconds, present after transmission is complete).

- **"TO ID"="D"—Device or "S"—Server:**

"SDATE"	(if present) DATE/TIME (in VA FileMan format) each time the STATUS changes, once a task starts dealing with the message.
"SDATE MM"	(if present) DATE/TIME (in MailMan format) each time the STATUS changes, once a task starts dealing with the message.
"STATUS"	(if present) Status of the message (usually present before, during, and after sending).

- **QL^XMXUTIL3(XMZ,XMFLAGS,XMAMT,.XMSTART,XMFIND,XMTROOT)**

  Get a list of the "latered" addressees of this message. Optionally, find the "latered" addressees that match a string. Gets a list (similar in format to that produced by LIST^DIC) of the requested "latered" addressees.

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).
XMFLAGS	(Reserved for future use.)
XMAMT	(optional) How many? Number—Get this many. *****—Get all (default).
.XMSTART	(optional) Used to start the Lister. The Lister will keep it updated from call to call.
("IEN")	Start *after* this latered addressee IEN. Continues from there, with each successive call, to the end. (Default is to start at the beginning.)
XMFIND	(optional) Find the "latered" addressees that match the string. (VA FileMan's FIND^DIC is used.) If XMFIND is supplied, then XMAMT and XMSTART are ignored; a complete list is always returned.
XMTROOT	Target root (closed) to receive the message list. (Default is ^TMP("XMLIST",$J).)

  Output Parameters:

.XMSTART	(As defined above.)
XMTROOT	Fields returned under XMTROOT for each "latered" addressee:
"TO NAME"	Latered addressee name.
"TYPE"	(if present) Addressee type: I—Information Only C—cc (Carbon Copy)
"BY DUZ"	DUZ of the person who "latered."
"BY NAME"	Name of the person who "latered."
"WHEN"	When will the message be delivered (in VA FileMan format).
"WHEN MM"	When will the message be delivered (in MailMan format: **dd mmm yy hh:mm**

- **QN^XMXUTIL3(XMZ,XMFLAGS,XMAMT,.XMSTART,XMTROOT)**

  Get the network header records from a message that originated at a remote site. Gets a list (similar in format to that produced by LIST^DIC) of the message's network header records.

  Input Parameters:

XMZ	Message IEN in the MESSAGE file (#3.9).	
XMFLAGS	(Reserved for future use.)	
XMAMT	(optional) How many?	
	Number—Get this many	
	*****—Get all (default)	
.XMSTART	(optional) Used to start the Lister going. The Lister will keep it updated from call to call.	
	("IEN")	Start *after* this line IEN.
		Continues from there, with each successive call, to the end.(Default is to start at the beginning.)
XMTROOT	Target root (closed) to receive the message list. (Default is ^TMP("XMLIST",$J).)	

  Output Parameters:

.XMSTART	(As defined above.)

Example:

```
>D QN^XMXUTIL3(978437)
>D ^%G

Global ^TMP("XMLIST",$J
        TMP("XMLIST",$J
^TMP("XMLIST",541073053,0) = 12^*^0
^TMP("XMLIST",541073053,1) = Received: from ISC-SF.VA.GOV by
MAILMAN.ISC-SF.VA.GOV (MailMan/7.1 Turn Around) id 978437 ; 1 May 1998
06:38:29 -0700 (PDT)
^TMP("XMLIST",541073053,2) = Received: from FORUM.VA.GOV by ISC-
SF.VA.GOV (MailMan/7.1 TCP/IP-MAILMAN) id 1204177 ; 11 Mar 1998
09:25:25 -0800 (PST)
^TMP("XMLIST",541073053,3) = Subject:Released DI*21*43 SEQ #39
^TMP("XMLIST",541073053,4) = Date:11 Mar 98 12:22 EST
^TMP("XMLIST",541073053,5) = Message-ID:<26463552@FORUM.VA.GOV>
^TMP("XMLIST",541073053,6) = From:<National Patch Module@FORUM.VA.GOV>
^TMP("XMLIST",541073053,7) = To: PUCE.FIL@FORUM.VA.GOV,
G.PATCH@FORUM.VA.GOV, G.SUPPORT@FORUM.VA.GOV, G.SUPPORT@ISC-
ALBANY.VA.GOV,
^TMP("XMLIST",541073053,8) =     G.SUPPORT@ISC-BIRM.VA.GOV,
G.SUPPORT@ISC-CHICAGO.VA.GOV, G.SUPPORT@ISC-DALLAS.VA.GOV,
^TMP("XMLIST",541073053,9) =     G.SUPPORT@ISC-SF.VA.GOV,
G.SUPPORT@ISC-SLC.VA.GOV, S.A1AE SERVER VERIFIED@FORUM.VA.GOV,
^TMP("XMLIST",541073053,10) =     PARKS.RICHARD@FORUM.VA.GOV,
NAPOLI.GERALD@FORUM.VA.GOV, HODGES.BRYAN@FORUM.VA.GOV,
^TMP("XMLIST",541073053,11) =     G.CSNATHD@FORUM.VA.GOV
^TMP("XMLIST",541073053,12) =
Global ^
```