Using *mcqueeno* MH

1.0.0 Why MH

The MH system is very different from most mail user agents. Instead of running one large program which handles all mail functions and keeps messages in one large file, MH is a collection of smaller single-purpose programs used to manipulate mail messages which are kept in individual files. MH may seem more complicated or harder to use than other mail systems, but MH has been designed to allow you to take full advantage of existing Unix commands and programs in connection with mail messages. For example, you can use your usual text editor, spelling program, and printer commands on individual messages.

1.1.0 Why *mcqueeno* MH

MH has been chosen for use to handle the *mcqueeno* mail queue for a number of reasons. First of all, it will allow multiple staff members to read and handle system mail at the same time, without need for locking a mailbox as a whole while one person is reading it. Secondly, it allows a very flexible message format that can be manipulated by locally developed tools for special purposes; for instance a uniform way of adding status lines to a message in the queue.

Other reasons include:

1) ability to export the system MH folders on an NFS partition to individual workstations, so MH can be actually run on the workstation but will operate on the global folder.

2) easily allows system mail readers to be added to the system without requiring root or special privileged access.

3) adds personal touch of system mail being answered by a specific user name 'joe' instead of 'root'. (or anonymously by "consult" or "*mcqueeno*")

4) and many, many more... [cures baldness]

2.0.0 Getting Started with *mcqueeno* MH

Here are the things that need to be done before you start as a *mcqueeno* MH user:

□ You will need an account on a machine with MH installed, the MH queue software/files installed, and which has filesystem access to the *mcqueeno* queue.

□ Get a systems person to add you to the group *mcqueeno*.

□ Edit your .login or .cshrc to include /usr/pkg/bin /arpa/gm/m/mcqueeno/bin commands in your path and to set the FOLDER environment variable to *mcqueeno*.

□ Login as yourself.

□ Execute "/arpa/gm/m/mcqueeno/mhsetup/install +mcqueeno". WARNING: this will overwrite your existing ~/.mh_profile and/or parts of ~/Mail/. You may want to do this by hand if you are already an MH user and do not want these things changed.

□ Begin using commands described in Section 3 to deal with *mcqueeno* MH.

3.0.0 Fundamental Commands in *mcqueeno* MH

3.1.0 Fundamental Concepts of MH

The way MH stores mail messages is much different than more common mail tools like /usr/ucb/mail, where your mail is stored as a single file (usually /usr/spool/mail/<*yourlogin*>). Instead, MH stores **each message as a separate file**. Groups of these files are called *folders*. What the folders are named and which messages are contained in them is decided by the MH user. Usually, the user's folders are stored in ˜/Mail. Within these folders, which are just run-of-the-mill directories, the mail messages are stored as standard text files with their file name being the message number in that folder. Old messages (ones that have been deleted) may be present with the name #<number>. These are easy to delete with a find(1) once a week or so.

There are other files in ˜/Mail that are important, too. You can find these described in section 3.3 of this document.

Just as an example, here is a listing of ˜trent/Mail:

```
aliases      cssun/      inbox/         outbox/       tag/
cm/          drafts/     incoming.log   repl.format   trap/
coggs/       evi/        marga/         sunbugs/      trash/
components   forwcomps   mhl.format     sysadm/       mcqueeno/
context      forys/      misc/          systems/      workshop/
```

The directories you see here are actually used as folders. Some of the other files will be discussed later.

Here is a listing of ˜trent/Mail/cm: (my "connection machine" folder)

```
1      11     13     15     17     3      5      7      9
10     12     14     16     2      4      6      8
```

Each of these files is an individual mail message. You could access these by using cat(1) or more(1), but as you will see later there are much nicer tools provided in the MH system.

3.1.1 Common folders in a *mcqueeno* MH user's Mail directory

Folder	Purpose
mcqueeno	active *QUEUE* mail
outbox	record of your outgoing mail
trap	all incoming *mcqueeno* mail (an archive)

inbox your personal incoming mail queue (if you use MH)

3.2.0 Commands used in *mcqueeno* MH

Here are the most commonly used commands in *mcqueeno* MH. For the most part, all other MH commands will work as well, but for simplicity only a few are listed here. In addition, not all options to all commands are presented here, do a 'man commandname' for the full option list.

comp

comp
This is used when **comp**osing a new mail message to be sent to someone. It puts you into your default editor with fields at the top to be filled in as to who the message goes to (To:), etc. Recipients are separated by commas. When you finish, exit the editor and it will prompt you with "What Now?". Usual responses to this prompt are edit, send, or quit. Type ? for help at this prompt.

Example of use: comp

folder

folder [+<foldername>]
This is used to either show what your current working folder is (with no arguments) or to change to another folder.

Example of use: folder +*mcqueeno*

folders

folders
This is used to show all available folders.

Example of use: folders

forw

forw [msgs]
This is used to **forw**ard a message to another user(s). It puts you in your default editor with the text of the current message included. If a message number is specified, it forwards that message. When you are done, you will be at the "What Now?" prompt.

Example of use: forw

glom

(local) glom [+<folder>] [-a glommed-message] message-list
This is used to combine messages from the *mcqueeno* folder ($FOLDER) into one message. When interpreting a message list from the command line, **glom** uses **pick** so that things like "1-6 9", "cur", and "last:6" are valid. The old messages are removed using **rmm** and the new message is placed into the lowest numbered message from the list of old messages. The -a option, or -add, is available so that adding more messages after the initial "glomming" does not make a hard-to-read mess of a message. The new message is addressed to the current folder, marked as being from the user doing the glom, and annotated using **anno** so that the message can be recognized as having been glommed. The "cc" line of the new message will contain all the mail addresses from the "from"

and "cc" lines of the original message. If "+<folder>" is specified, the actions are done on the specified folder. Glom will work on any folder, so be careful. The +folder option is intended for those users that access more than one system MH queue.

usage:
    glom -h[elp]
    glom message-list
    glom -a[dd] glommed-message message-list

Example of use: glom 1-9 10 last:5

next

next
This advances to the next message in the folder and displays it with **show**.

Example of use: next

notack

(local) notack [+<folder>]
This command displays messages in the *mcqueeno* folder of which the sender has not received an acknowledgement of receipt. (see **stat**) The +<folder> option works as in **glom**. **notack** is meaningless when used on folders other than system queues like *mcqueeno*.

Example of use: notack

prev

prev
This displays the previous message in the folder using **show**.

Example of use: prev

refile

refile [msgs] +<foldername>
This is used to moved messages between folders. The messages specific are moved to the folder <foldername>. If no message is specified, the current message is refiled.

Example of use: refile +misc

repl

repl [msg]
This is used to reply to a message. If no message is specified, the current message is assumed. It puts you into your default editor with the text of the message usually included and offset from the left margin. When you are done, you will be at the "What Now?" prompt.

Example of use: repl

res

(local) res [+<folder>] [msg]
This command is used when you are ready to resolve a *mcqueeno* message. It

works very similar to **repl**, except that it always *moves* the message to a temporary file (for locking) and automatically carbon copies mcqueenomr, ops. The +<folder> option works as in **glom**. The original message is *not* returned to the *mcqueeno* folder (since it has been resolved).

Example of use: res /* to resolve current message */

rmm

rmm [msgs]
This is used to delete a message from a folder. If no message is specified the current message is assumed.

Example of use: rmm

scan

scan [+<foldername>]
This is used to get a summary of the messages in a folder, usually displays date, subject header, etc. If no folder is specified current folder is assumed.

Example of use: scan

show

show [msgs]
This is used to display a message. If no messages are specified current message is assumed.

Example of use: show 17

stat

(local) stat [-e] [+<folder>]
This is used to add status information or to acknowledge to the current *mcqueeno* message. Will place you in **repl** with

----
Status: by <yourusername> <date>
----

at the end. You should type the acknowledgement or change in status message at the end of the file (after the Status: lines). Also, edit the recipients in the To: and Cc: lines of the message to reflect the desired addresses. Then, upon sending the message, the text you typed after the last Status: field will be appended to the end of the original message in the *mcqueeno* folder.

The +<folder> option works as in **glom**.

When used as stat -e, you are able to edit the message in the folder itself, but it also appends a Status: line at the end of the message for you to include your comments after. The message is not forwarded to send after using stat -e. Warning: do not delete the guts of the message using stat -e, as there is no other copy of the message in the folder.

Example of use: stat
stat -e

3.2.1 Quick reference to *mcqueeno* MH

Command	Purpose
comp	compose a new message to a user
dist	distribute a message to addtional people (see dist(1))
folder	show current folder or change folders
folders	show available folders
forw	forward a message to additional user(s)
glom	combine *mcqueeno* messages of similar topic (local)
inc	incorporate your personal mail spool (see sec 6.0.0)
next	display next message in folder
notack	scan those *mcqueeno* messages that have not been acknowledged (local)
prev	display previous message in folder
refile	move message to another folder
repl	reply to a message
res	resolve a *mcqueeno* message (local)
rmm	remove a message
scan	display summary list of folder
show	display a message
stat	add status information to a *mcqueeno* message, acknowledge, or edit (local)

3.3.0 Other *mcqueeno* MH files

Here are just some of the other files that MH uses to deal with the preferences of the user. For the most part, there is no need to change any of the files that we have provided for you in *mcqueeno* MH. But, just in case you're curious... We strongly suggest you use the files we have provided.

~/.mh_profile

This file contains information about your MH environment, such as your default editor, what options you want to always use with particular commands, etc. The default *mcqueeno* MH .mh_profile looks like:

```
Path: Mail
Editor: /usr/ucb/vi
Msg-Protect: 600
Folder-Protect: 700
showproc: /usr/pkg/bin/mhl
repl: -filter repl.format -nocc me -nocc to -fcc +outbox -annotate
forw: -form forwcomps -annotate
dist: -annotate
comp: -form components
send: -alias aliases
```

~/Mail/aliases

File to contain your personal mail aliases.

~/Mail/components

File that contains the 'blank' message **comp** will use when creating a mesage.

~/Mail/context

File that contains the record of your current folder, as well as what your current message is in 'remote' folders (like *mcqueeno*) Don't edit this file.

~/Mail/forwcomps

This file describes how a forwarded message will look when using **forw**.

~/Mail/mhl.format

This file describes how what format will be used when you display a message with **show** (or equivalent). If you want to change anything, it probably should be this.

~/Mail/repl.format

This file describes what format will be used with replying to a message with **repl**.

4.0.0 Using *mcqueeno* MH

This is a basic procedure you should use when reading *mcqueeno* mail:

```
% notack        /* to list new unacknowledged messages */
% show #        /* to display the messages */
% stat #        /* to acknowledge or provide status on the message */
% show #        /* to display another message */
% res #         /* to resolve the message */
```

notack should be used to list a summary of all the messages that have not yet been acknowledged at least once. Use **scan** to list all the messages in the queue or a specific message-list.

stat should be used when the request or question posed in the message requires further action.

res should be used when the message no longer requires action and can be removed from the queue.

glom should be used to combine queue messages when there are multiple messages in the queue of the same or similar topic where the same action is required for them all.

4.1.0 General guidelines you should follow in *mcqueeno* MH

• All messages should AT THE VERY LEAST be acknowledged with the **stat** command within 24 hours.

• All un-resolved messages should be re-acknowledged and/or new status message appended after every 36 hours with **stat**.

• Don't answer a message that you don't know the answer to. ASK!

• Always cc: mcqueenomr on all replies. (**res** and **stat** do this automatically)

4.2.0 Things not to do in *mcqueeno* MH

• **PLEASE DO NOT** do operations on the main *mcqueeno* folder that affects the contents of the folder as a whole. For example, do not ever do a **sortm** or **packf** on the folder.

• Do not use **repl** or **rmm** when **stat** or **res** are more appropriate.

5.0.0 Other MH tools you might want to try

If you don't like the interface provided by the commands listed above, there are many other tools available for dealing with MH folders. Some that you might want to try:

xmh
gnu-emacs mh

If you use **xmh**, note that it does not gracefully handle the case where it tries to display a message it thinks is there but has been removed with a command executed outside of xmh (**res, stat, glom, rmm, refile, etc**).

5.5.0 Available standard MH commands

Here are the 'standard' MH commands.

```
anno (1)        - annotate a message
burst (1)       - explode digests into messages
comp (1)        - compose a message
dist (1)        - redistribute a message to additional addresses
folder (1)      - set/list current folder/message
folders (1)     - list all folders
forw (1)        - forward messages
inc (1)         - incorporate new mail
mark (1)        - mark messages
mhl (1)         - produce formatted listings of MH messages
mhmail (1)      - send or read mail
mhook (1)       - MH receive-mail hooks
mhpath (1)      - print full pathnames of MH messages and folders
msgchk (1)      - check for messages
msh (1)         - MH shell (and BBoard reader)
next (1)        - show the next message
packf (1)       - compress a folder into a single file
pick (1)        - select messages by content
prev (1)        - show the previous message
prompter (1)    - prompting editor front end
rcvstore (1)    - incorporate new mail asynchronously
refile (1)      - file messages in other folders
repl (1)        - reply to a message
rmf (1)         - remove folder
rmm (1)         - remove messages
scan (1)        - produce a one line per message scan listing
send (1)        - send a message
show (1)        - show (list) messages
sortm (1)       - sort messages
vmh (1)         - visual front-end to MH
whatnow (1)     - prompting front-end for send
whom (1)        - report to whom a message would go

mh-alias (5)    - alias file for MH message system
mh-format (5)   - format file for MH message system
mh-mail (5)     - message format for MH message system
mh-profile (5)  - user customization for MH message system

ap (8)          - parse addresses 822-style
conflict (8)    - search for alias/password conflicts
dp (8)          - parse dates 822-style
install-mh (8)  - initialize the MH environment
post (8)        - deliver a message
```

6.0.0 Using MH to manage your own mail

Using MH to manage your own mail spool is even simpler than using *mcqueeno* MH! Use **inc** each time you want to incorporate your mail spool (/usr/spool/mail/???) into your mh folder. The default folder these messages will be placed in is *inbox*. If you want, you can have this procedure done for you automagically (like *mcqueeno* MH) by using mhook(1).

Standard MH commands like **scan, show, repl, forw, rmm, comp** will work fine on your own mail spool. Commands that you probably want to avoid using on your own mail folders include **res** and **stat**.

Here is a script of a simple login/MH mail reading session:

```
anchor 1 >inc

Incorporating new mail into inbox...

 132  10/06 Brian Walker       Re: ST Keyboard<<In article <3153@ucdavis.ucdavi
 133  10/06 Andreas Girgensohn Change of group id<<I would like to be in the gr
 134  10/06 Brent Browning     Re: Yet another reminder about reliable message

anchor 2 >show 132

(Message inbox:132)

Date:    Thu, 6 Oct 88 13:51:28 MDT
To:      trent@boulder
From:    walkerb@tramp.Colorado.EDU (Brian Walker)
Subject: Local news posted


Return-Path: walkerb@tramp.Colorado.EDU

----------------------------------------------------------------

In article <3153@ucdavis.ucdavis.edu> kerchen@iris.ucdavis.edu (Paul Kerchen) w
rites:
>In article <1057@orion.cf.uci.edu> broode@orion.cf.uci.edu (Brian Roode) write
s:
>>Does anyone know if Atari sells replacement keyboards for the 520 ST?
>>
>Along similar lines, does anyone know if there is a detachable model
>of such a beast if it exists.  Sure would be nice....

actually, such a thing is not all that difficult.  The current keyboard has
been implimented with a serial interface.  If you connect the wires inside the
machine to an appropriate receptacle (the kind used for telephones), my guess
is that you could easily use a keyboard for a Mega ST.

Brian Walker, University of Colorado at Boulder||printf("Say please:] 0);
              walkerb@tramp.colorado.edu=======||if (say_please(user))
{ncar,nbires,sunybcs}!boulder!tramp!walkerb====||  be_nice(random());

anchor 3 >rmm

anchor 4 >next

(Message inbox:133)

Date:    Thu, 6 Oct 88 13:55:05 MDT
To:      trent@sigi
From:    Andreas Girgensohn <andreasg@sigi>
Subject: Change of group id
```

```
Return-Path: andreasg@sigi

--------------------------------------------------------------
I would like to be in the group 'fischer' instead of 'cs'.

Andreas

anchor 5 >scan

   1  09/19 John Foster        How can I get back issues of comp.sources.misc?<
   2  09/20 fake for net distr Updated local aliases with the following<<icarus
   3  09/20 Sylvan Ruud        What kind of name...<<is 'anchor' for a home mac
  71  10/05 Robert Coggeshall  Re: Re: how about some systems staff uniforms?<<
  81  10/06 Operator           Output from "cron" command<<Your "cron" job prod
 133+ 10/06 Andreas Girgensohn Change of group id<<I would like to be in the gr
 134  10/06 Brent Browning     Re: Yet another reminder about reliable message
 135  10/06 The devil himself  Local news posted<<Path: boulder!daemon From: ad

anchor 6 >logout
```