MJDLM: Majordomo based Distribution List Management

"It's not spam when WE send it"

Vincent D. Skahan, Jr. and Robert Katz – The Boeing Company

ABSTRACT

In early 1998, we were asked by Corporate Communications to develop a facility for providing a subscription based internal company mailing list capability that would permit Senior Executive management to send messages on an irregular basis from anywhere in the world.

Shortly thereafter, we were also asked to provide a one-use-only mechanism to send all 175,000 worldwide employees a (self-qualifying) message of where internally to report perceived spam in order to support corporate efforts to reduce incoming spam through technical means. Much to our surprise, delivering this message was a technical non-event.

The success of these efforts led Corporate Communications to request a more general system for permitting multiple mailings to targeted audiences, with a goal of completely eliminating the paper-based communications Management Information Bulletin systems, hopefully at considerable cost savings.

Given time and budget constraints, we chose to base our solution on majordomo.

This paper describes a scalable method for handling deliveries to multiple majordomo mailing lists with a minimum of administration. Ancillary issues such as sender authentication, message constraints, bounces, mail replies, and mailing list recipient management are also described.

This system is in use daily in Boeing and has easily supported lists as large as 150,000 recipients – drawn from custom SQL queries of the company's employee database. Conservative estimates of the savings in moving to a fully electronic communications mechanism exceed $1,000,000 per year with cycle time that has improved from several days to only a few hours.

Introduction

Beginning in January 1998, we were asked by Senior Executive management to provide a subscription based internal company mailing list that would be used for sending messages on an irregular basis.

Since the messages were coming from Senior Executives, there was a feeling that (of course) everyone in the company would ultimately choose to subscribe, making the potential audience 175,000 employees.

The only way to implement what was requested in the required time frame was to try to leverage our extensive experience with majordomo[1] and to develop a custom set of tools and processes that could be wrapped around majordomo. But scale was an obvious concern.

While majordomo has been in use within Boeing since 1992, usage profiles tended to be many reasonably small volume, small size (under 1000 recipient) lists. An Internet search of the majordomo-workers list archives appeared to show that majordomo had been proven only to approximately 25,000 recipients even when adding such optimizations as bulk-mailer.

We were talking about 'moving the decimal point' in terms of scale, and doing so for the highest levels of Senior Management. Things seemed bleak.

A Fortuitous Request

Shortly thereafter, we were asked to actually send a special "help us stop incoming spam" message to all e-mail enabled employees. We chose to use this one-time-only message as a test of how majordomo scaled within our existing hub-based electronic mail environment, which at that time supported well over 170,000 potential recipients.

(To say the least, all involved on the technical side quite enjoyed the irony of the message text.)

A team of internal majordomo, electronic mail, network infrastructure, and mail hub experts was put together to develop a plan to do a staged test with the appropriate instrumentation to see where things would begin to "brown out."

In addition, we set up the majordomo lists as exploder lists made up of 2000 recipient sublists, so that when things eventually failed, we could resend the messages more effectively to the victims.

Much to our surprise, even the largest (90,000 user) list was delivered with no problems, with the messages leaving the majordomo system in 5.8 minutes and getting through the mail hub (where readdressing to explicit smtp address occurs) and out to the variety of destination systems (Exchange, UNIX SMTP, CC:Mail, etc.) in under 2.5 hours (see Figure 1).

A project was born.

Project Background

The Distributed List Management Service (DLMS) project was launched in mid-1998 to accomplish the following objectives:

- Support the requirements of Senior Executives to quickly, effectively, and completely disseminate information to ad hoc subsets of Company employees.
- Have the service work through the existing Public Relations and Communications Focals
- Adhere to the concept of alerting with E-mail and informing via the Web.
- Transition company-wide communications from a paper mail to an E-mail/Web basis
- Give the message author total control over (and responsibility for) message content.

A multi-disciplinary (8-person) team was assembled from database, mailing list, mail systems, communications and Public Relations specialists. The project had Vice Presidential sponsorship and produced the first working list within six weeks.

A message was sent to that list (10,331 recipients) on July 17, 1998. The Service was put in full production on April 24, 1999 with 65 active mailing lists.

MJDLM Requirements

There was considerable negotiation regarding what we would and would not support, and what constraints would be defined for the system and users.

Types of Lists

We envisioned two types of lists, temporary (one use), and permanent (more than one use). Temporary lists were intended to be disabled one week after use.

Message and List Limitations

Messages would be limited to majordomo's default 40,000 character limit in ASCII only.

Users were (very strongly) requested to limit their messages to brief pointers to web pages, in order to limit network and system load as much as possible.

Authors were fully and completely responsible for their content.

We agreed not to try to stop any message after the approved sender hit the "send" key.

All lists would be set up within majordomo as "restrict_post," "private," with moderation and advertising off.

Assembly of the Recipient List

It was decided that recipient lists would be created by SQL queries of the corporate personnel database, and would be updated weekly. Majordomo would then blindly believe that the SQL query and resulting employee e-mail addresses it returned from the corporate databases were complete and accurate.

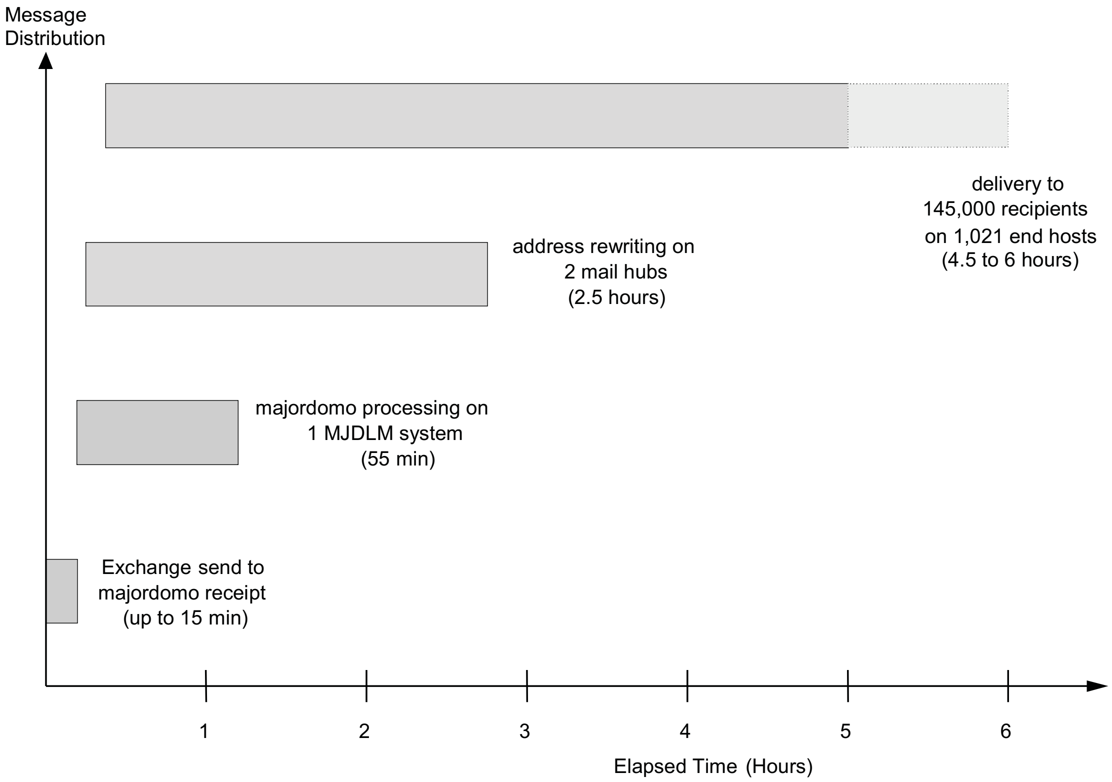

Figure 1: Typical MJDLM message delivery flow and latency.

Each REXX script (see Figure 2) would sort the recipients by the host portion of their Email address and produce a list of recipients which is suitable to FTP to the Majordomo Server. A separate REXX script for each list would also produce the number of employees with no E-mail who would otherwise fit the criteria.

Prior to actually using the data, a "sanity check" program would be run in advance to ensure proper address format, uniqueness, and continuity versus the previous week's list.

The costly address rewriting from "stable" to "explicit" e-mail address would take place on the corporate mail hubs to effectively serve as a throttle to prevent system and network "brown out."

Leveraging the pre-existing multi-hub architecture and associated load-levelling removes the need for special sendmail, majordomo or operating system tuning [3,5] on the MJDLM systems themselves that would have been necessary if the Majordomo system had to do the actual name resolution and delivery itself.

In addition, it permitted us to use whatever existing (weak) hardware we could acquire internally without the delay and stress of having to procure additional hardware for an unproven implementation.

Security and Network Considerations

Any message sent to more than 30,000 recipients would automatically notify those in charge of the mail hub(s) and DLMS agents.

Headers would be rewritten only by majordomo's resend program.

All MJDLM lists would have a restrict-post file of authorized senders and an explicit Reply-To: field.

Only the majordomo userid would be permitted to run the MJDLM program and modify list data.

The system needed to function in production on a 7 day x 24-hour basis.

Errors, Traceability and Message Archiving

Each message sent would be archived for a maximum of one year, but there was no requirement to permit retrieval of archived messages (majordomo's archive2.pl or the like).

Bounced messages would be the responsibility of the List-owners, with support from messaging specialists as needed.

Currently (by policy), all lists are owned by MJDLM project personnel rather than the various corporate communications focals, with one mailbox getting all the incoming bounces from all lists.

Bounces in the form of undeliverables, mail failures, host unknown, user unknown, warnings, and the like are collected in the list-owner's Mail subdirectory and managed by the procmail program via extremely simple recipes (see Figure 3).

These bounced messages help to verify certain recipient's complaints about not getting the message and wanting to. (There are also complaints about getting the message and not wanting to.)

Messages sent to the mailing list by unauthorized senders would be permitted to bounce for approval to the list owner, where they would be silently (by policy) ignored.

```
# To list all employees:

    BEMS_EMPL_ID        NE      ''      # Employee number exists and
    EMAIL_ADDR          NE      ''      # E-mail address exists and
    PEOPLE_TYPE         IS      E       # Is a Boeing Employee and
    EMPL_STAT_CD        IS      A       # Is on Active Status

# To limit to Puget Sound, WA Engineering Managers add:

    TBC_LP_7            EQ      S       # Is in Puget Sound and
    ENTITY_ID           EQ      TBC     # Is in Heritage Boeing and
    PAY_CAT             EQ      M       # Is a Manager and

# Where specific Budget Organization names match the 1st character or
# 1st 3 characters as indicated for Engineering

    WHERE ((ORGN_STRUC EQ '4FT9')
OR (EDIT(TBC_BUDGET,'9') EQ 'B' OR 'E' OR 'U' OR 'R' OR 'N')
OR (EDIT(TBC_BUDGET,'999') EQ 'MAB')
OR (EDIT(TBC_BUDGET,'999') EQ 'MGB')
OR (EDIT(TBC_BUDGET,'999') EQ 'MHB')
OR (EDIT(TBC_BUDGET,'999') EQ 'MMB')
OR (EDIT(TBC_BUDGET,'999') EQ 'MTB'))
```

Figure 2: Example SQL Query Criteria for Targeted Lists.

Replies sent to the list instead of the Reply-to: address would be stopped by the restrict_post configuration, but the List-owner would manually forward it to the message sender as a courtesy.

How It Works From the User Perspective

Figure 4 shows a process data flow of list construction and handling, message sending and handling of returned mail.

The list is created after a form is filled out and approved by one of 13 company-wide communication focals. The criteria are researched (e.g., All Engineering Managers in Puget Sound, WA) and the SQL query developed to produce the recipients of the list.

Test messages from bonafide addresses are sent to verify the information on the form. When the list is ready, it is created using the MJDLM commands (see Appendix 1).

An E-mail letter is sent to all who are permitted to send to this list indicating the list name and address to send messages, the criteria, the number of recipients with and without E-mail, and orienting rules of use.

Lists are updated on Monday mornings, after weekly corporate database updates occur on the previous Friday evenings. Lists get disabled if there have been one-time messages sent to temporary lists or if the list becomes obsolete due to organizational makeovers.

```
:0
* ^Subject: Returned mail: User unknown
bounces.user-unknown
:0
* ^Subject: Returned mail: Host unknown
bounces.host-unknown
:0
* ^Subject: Returned mail: Local configuration error
bounces.cfgerror
:0
* ^Subject: Returned mail: Non delivery
bounces.nondelivery
:0
* ^Subject: Returned mail: Too many hops
bounces.hops
:0
* ^Subject: Undeliverable
bounces.undeliverable
:0
* ^Subject: Message status - undeliverable
bounces.undeliverable
:0
* ^Subject: unable to deliver
bounces.misc
:0
* ^Subject: Returned mail
bounces.misc
:0
* ^Subject: Non-delivery Report
bounces.delivery-reports
:0
* ^Subject: Delivery-Report
bounces.delivery-reports
:0
* ^Subject: Delivery Notification
bounces.delivery-reports
:0
* ^Subject: DELIVERY FAILURE
bounces.delivery-failure
:0
* ^Subject: Warning
bounces.warnings
:0
* ^Subject: Mail failure
bounces.mailfailure
```

Figure 3: Procmail Recipes.

The philosophy is to do the updating to all lists at once, rather than handle each explicit list individually. Other list handling is done on an exception basis.

When a message is sent to a mailing list, it goes out unmoderated. If the restrict-post addresses are not matched, it bounces to the list owner. As a courtesy, the sender is called to indicate this situation, since the sender may not necessarily be qualified to receive the message. Other bounces are ignored. Usually, the sender resends the message from the correct mailbox.

The major senders of weekday messages to all employees with E-mail are the Company electronic newspaper (daily) and the CEO's message of the week. Some other messages are timed to be released at a certain time of day for delivery purposes due to their media importance.

For the "all employees" list consisting of 144,561 recipients on June 30, 1999 there were 566 bounces in all categories yielding a return rate of 0.39% (see Figure 5). This is purely a function of company employee database and DNS accuracy.

Hardware Considerations

The MJDLM software and associated data runs on two Sun Sparcstation-5 systems with 128 MB RAM and one 2GB Hard Disk each.

One system serves as the primary server, with an MX record and associated majordomo and sendmail configurations to permit the backup to take over automatically if the primary production server is unavailable.

Both systems get full production monitoring and support on a 24x7x365 basis.

The actual MJDLM data is updated identically on each server via the MJDLM "update" functionality rather than updating the primary server and then synchronizing the backup server through other possible means such as rdist or mirror.

Since the throttle in the entire system is the time necessary to rewrite the addresses on the mail hubs, using slower than possible majordomo systems was not a problem, and served as a considerable cost savings since the hardware was in-house.

Centralizing mail delivery to occur by the mail hubs also permitted us to leverage the existing excellent logging and problem resolution capabilities already present there.

Why Majordomo?

We selected majordomo as the (hidden) list exploder subsystem mainly due to our existing experience with the product, our need to restrict postings to a very small set of author addresses, and the known behavior of how it permits easy header modifications (Reply-To: and the like).

We also had a pre-existing implementation in one small corner of the company that used a similar database query to majordomo model to routinely generate and update org-chart based distribution lists.

While this previous implementation was on a much larger number of lists (over 400) but far smaller

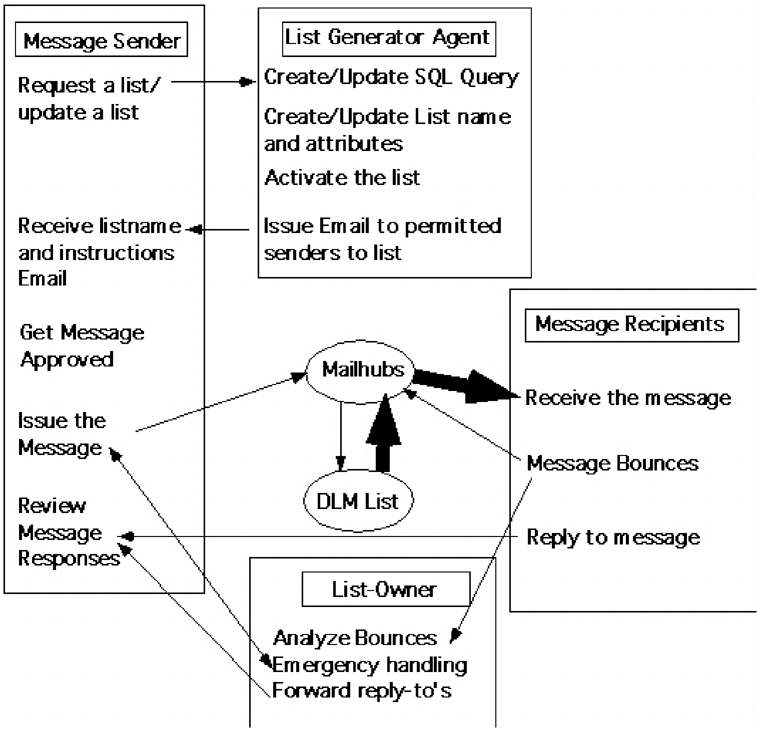

Figure 4: Basic Process and Message Flow.

number of recipients per list (under 1000) scale, we had five years of experience saying that the basic architecture was sound.

Lastly, the usual time and budget constraints prohibited acquiring, learning, and implementing any of the various commercial packages that claim to be able to support lists that could approach 200,000 recipients in size.

Custom Code Description and Risks

Our most important design point is that "*we will not modify* majordomo *in any way*." This permits us to leverage the known behavior of majordomo without having to "roll our own."

This decision has caused us to turn down some user requests for features that are contrary to majordomo's implementation or general philosophy.

For example, users who tend to forget to send from approved mailboxes asked for a modification to return a "you did it wrong this way" message to them rather than a (hidden) bounce for approval to the list owner.

Internally, the MJDLM software is written in object oriented perl that uses a multilevel Data::Dumper structure containing all the information that needs to be stream edited into the per-list majordomo configuration files, or that needs to be saved, to permit adding, updating, disabling, and restoring a list (see Figure 6).

All installation-configurable parameters are present in external config files, with almost all routines in perl modules for reuse. There's little reusable content in the MJDLM perl program itself.

Lists are created on the majordomo side by the equivalent of sed on a template majordomo config file. This has a number of associated risks that we felt were acceptable for the foreseeable future:

- We're hard-wired to a particular version of majordomo. The major upgrade to majordomo 2.0 will cause a significant rewrite of much of the internal implementation.
- We must ensure that the config files we write are absolutely perfect, as we go around majordomo's existing "writeconfig" sanity checks.
- We have to get the template files correct right away due to poor inherent ability to change the template config and "make it so" on existing lists. We currently must disable/enable each list to cause a rewrite according to the current template.config.

All lists use the same template majordomo configuration file, although we support use of custom per-list configuration file templates if needed.

System Tuning

We have done nothing to tune sendmail or the operating system in any way, relying on sendmail 8.x and Solaris 2.6 to do the right things.

Number of recipients			144,561
No E-mail			45,860
Total That Matched Criteria			190,421
procmail filter files			
bounces.delivery-failure			15
bounces.delivery-reports			12
bounces.host-unknown			21
bounces.misc			36
bounces.slvma			9
bounces.undeliverable			416
bounces.user-unknown			12
bounces.warnings			0
list owner filtered Bounces Subtotal			521
Bounces to reply-to address			39
	delivery-failure	35	
	undeliverable	1	
	user-unknown	3	
other bounces to list-owner			6
	delivery-failure	2	
	user-unknown	4	
Total Number of Bounces			566
Total % of Bounces			0.3915%

Figure 5: Bounce Analysis for All Employees List June 30, 1999.

Since all the address rewriting of employee e-mail addresses from ''stable public address'' to ''current real address'' happens on the mail hubs, it really doesn't matter how fast the message clears the majordomo host as the bottleneck is the address rewriting on the mail hubs.

Majordomo is minimally tuned for even the largest lists. Lists under 2000 recipients are unaltered in any way. Lists over that size are configured as ''exploder'' lists consisting of 2000-recipient sublists.

While this tends to agree with some of the published documentation [5] regarding parallelizing delivery to large lists, in our particular case, the value originated from blind fear.

The value 2000 represents twice the largest majordomo list we had run in production, and approximately 10% of the size of the largest list we could find mentioned in the majordomo-workers archive.

The thought was that if a delivery ''broke,'' we would be able to determine which sublists were hung in the queue, and manually re-send the message to just those recipients if needed. Fortunately, this hasn't happened yet.

Optimizing Delivery Time

While the majordomo system can run safely unoptimized, delivery time to the end users is an important success metric for the project.

We originally structured the SQL queries to generate output sorted by recipient employee messaging system unique ID number (optimized for the human, not the computer).

Since the employee database knows both the unique ''key'' for the employee, and the ''value'' of the e-mail address where that user ultimately reads their mail, it was suggested by the mail hub personnel to alter the SQL reports to sort based on the hostname the recipient mail really goes to.

This got us a 55% decrease in number of messages required to be delivered and indirectly a reduction in total delivery time (ala bulk mailer) by letting the mail hub deliver to the destination addresses in a streaming mode.

```
List::new (code admittedly derived from Perl5 Camel p298)

=item new()

This creates a List object with the contents initially
undefined. It is up to the calling program to populate
the 'pieces' of a List object.

=cut

%fields = (

     # the following items are specified by the users

     name              => undef,   # domo list name
     owner             => undef,   # domo owner e-mail address
     description       => undef,   # domo description string
     reply_to          => undef,   # follows domo reply_to values
     restrict_post     => [],      # e-mail of approved posters

     # the following items are calculated

     num_sublists      => undef,   # number of 'sub-lists' needed
     list_type         => undef,   # temporary or permanent
     isa_sublist       => undef,   # am I a sublist ?
     staged_recipients => undef,   # filename of last recipient list
     num_recipients    => undef,   # number of recipients
    );

sub new {
  my $that  = shift;
  my $class = ref($that) || $that;
  my $self  = { _permitted => \%fields, %fields, };
  bless $self, $class;
  return $self;
}
```

Figure 6: Perl Code Example.

A total of 10 MS-Exchange bridgeheads receive mail for 91% of the employees with E-mail addresses. Approximately 1,000 other systems receive the remaining 9% of the messages.

The lower number of end messages from the mail hubs also helps protect the network from overloading.

Use To Date

Between July 17, 1998 and June 30, 1999, there have been a total of 1278 messages sent to 55 lists, with 38 lists having yet to receive their first message. The average number of messages sent to all lists is 106.5 per month.

Of the 1278 messages sent, the three largest lists (currently averaging 149,802, 72,258, and 34,730 recipients each) have processed 155 or 12.13% of the messages sent while they generated 87.32% of the actual traffic (or 21,833,519 messages) of the total 25,003,407 discrete message deliveries to date (see Figure 7). Since the number of employees has been declining since the beginning of this year, this represents a lower bound value.

The top three lists together issued 1.6 messages per weekday for June 1999.

Web Data Interface

An independent adjunct to this project is the use of a daily updated internal Web site that provides a List Request or List Change form, the status of the Server (if down), the status of lists both existing and gestating, and a variety of database driven reports that map sender names to permitted sender Exchange Group mailbox names. Focals and their associated existing lists, and Reports for each list indicating the permitted sender and other attribute information for that list are also viewable.

This permits the information flow to be centralized and authoritative for project members, senders and interested parties. Much of this data is password protected at this time and disclosed only to Focals and project members.

Benefits

MJDLM provides a new capability to perform an ad hoc mailing list distribution based on a Company Employee database that is refreshed weekly. This permits targeted individual messages as well as message-series to be sent on a regular basis.

Annual cost savings are estimated to be $1,000,000/year in reduced or eliminated paper printing costs, with additional improvements in cycle time by permitting real-time creation and distribution of time-critical communications by using Internet Standard Time for message creation, approval, sending and distribution rather than internal or external ‘‘Snail Mail.’

Risks and Concerns

There is concern that as people get accustomed to the service, their expectations might start to exceed the infrastructure (or the recipients’) capacity to absorb such broadcasts.

	Messages per Month	Messages per weekday
Jan-99	17	0.85
Feb-99	20	1.00
Mar-99	28	1.22
Apr-99	32	1.45
May-99	26	1.38
Jun-99	32	1.60

Large List Name	Average Size 6/30/99	Messages Sent by 6/30/99	Messages Received by 6/30/99	Estimated Bounces by 6/30/99
All Employees	149,802	141	21,122,131	82,700
All Commercial Airplanes	72,258	6	433,547	1,697
All A/C & Missile Systems	34,730	8	277,841	1,088
Subtotal for Large Lists		155	21,833,519	85,485
Total All Lists		1278	25,003,407	97,896
% of Large Lists to All Lists		12.13%	87.32%	

Figure 7: Frequency of Use For Large Lists (over 30,000 recipients).

Fortunately, the Company Focals who decide who can have a list created and what it can be used for have been judicious in their creation and use of "big" (over 30,000 recipient) lists and helpful in limiting their use of big lists to times outside Puget Sound prime-time.

The leading use of the MJDLM system in terms of number of delivered messages has been in presenting a new electronic "Boeing News Now" daily to all employees, containing short news briefs thought to be of interest to all employees, with associated pointers to the appropriate web pages. This service broadcasts five days a week at 9:00 p.m. Seattle time.

Especially with the creation of an all employees list, there were initial fears of mass resistance to being "spammed" by a daily news service. It turned out that relatively few wished to publicly express the desire to opt out of receiving these mailings. Those that had religious views (interestingly enough, which included much of the project team) were able to filter out the messages on their own.

At this time, however, *only unemployment or a mistake in the Company Employee database prevents a recipient from being included on the various targeted distribution lists*. This appears to be an unalterable political reality at this time.

Futures

Internally, as we are "overrun by success," we believe the one file per list database structure we're using internally will need to be updated to a multilevel DBM or LDAP configuration for scalability.

Our command-line interface is admittedly minimal, and enhancement with a GUI or web front end [2,4] is needed for usability.

Even with a relatively low bounce rate, we have encountered situations where real-time use of procmail to filter incoming bounces has caused system stability problems. We are considering filtering bounces in batch mode [3] rather in real-time via procmail or similar tools.

There are also plans to better integrate with the continuing evolution of Company databases over the next few years and to fully automate the current (largely manual) status-related web pages.

In the meantime, there will be continued study and fine-tuning of a mailing system that all levels of corporate management have embraced and are beginning to rely on as part of their daily business processes.

Acknowledgements

The authors wish to acknowledge the following people who influenced the quality and fine-tuning of MJDLM.

They include Mary Jo Shipley, Rick Willard, Laurie Thompson, Dean Richardson, Sean O'Sullivan, Paul Swortz, Don Meyer, Brenda Kern, Patrick Palmer, Denise Sirven, Jim Crozier, Ron Hanson, Karen Burt, and Don Fitts.

Author Information

Vince Skahan <vince.skahan@boeing.com> is a System Design & Integration Specialist at the Boeing Company. He holds a B.S. in Chemical Engineering from Drexel University, and has been doing large scale unix administration since 1987. He's run majordomo since immediately after learning about it at the 1992 LISA conference, and also admits to being the original author of the majordomo FAQ.

Robert Katz <robert.katz@boeing.com> is a System Design & Integration Specialist at the Boeing Company. He holds an M.S. (Math) degree from Polytechnic Institute of Brooklyn. His technical interests include Large Scale E-mail communication, UNIX, Perl, Transaction Processing Monitors and CORBA. He teaches UNIX: Introduction, Shell Programming, and System Administration at Highline Community College in Des Moines, WA in his spare evenings.

References

[1] Chapman, B., "Majordomo – How I Manage 17 Mailing Lists Without Answering '-request' Mail," *Proceedings of the Sixth Systems Administration Conference (LISA VI)*.

[2] Viega, J., et al., Mailman: The GNU Mailing List Manager, http://www.usenix.org/publications/library/proceedings/lisa98/viega.html .

[3] Rose Chalup, S., et al., "Drinking from the Fire(walls) Hose: Another Approach to Very Large Mailing Lists," http://www.usenix.org/publications/library/proceedings/lisa98/chalup.html .

[4] Houle, B., "MajorCool: A Web Interface To Majordomo," http://www.usenix.org/publications/library/proceedings/lisa96/bhoule.html .

[5] Kolstad, R., "Tuning Sendmail for Large Mailing Lists," http://www.usenix.org/publications/library/proceedings/lisa97/21.kolstad.html .

APPENDIX 1: Example List Creation Transcript.

```
# mjdlm --help
 Version 2.09 usage:  mjdlm [-options] [-list LISTNAME] [-misc_actions]
 The following options operate on all lists, unless a particular list
 was specified (which restricts the action to the specified list only):
 --------------------------------------------------------------------
  -add             = add all lists needing adding
  -update          = update all lists needing updating
  (none specified) = do all required additions/deletions/updates
 Miscellaneous options include:
 ------------------------------
  -help      = give help message
  -debug     = run in debug (but not execute) mode
  -verbose   = verbose output
  -prompt    = prompt user (used to create a new control file)
  -show      = show control file for a particular list
  -showlists = show what lists currently exist
 The following options REQUIRE a list to be specified:
 -----------------------------------------------------
  -temporary = convert the list to temporary status
  -permanent = convert the list to permanent status
  -restore   = restore the list from the ‘trash’
  -rename    = rename a list (and its sublists)
  -disable   = disable a list (and its sublists)

# mjdlm -prompt
     Enter the desired list name.  Majordomo lists need to be
     lowercase, with no embedded spaces or ‘‘funny’’ characters other
     than [a-z], [0-9], and perhaps "+", "_", and "-"
list name [] : mytestlist
    Enter the desired reply_to configuration.  The choices are
       $SENDER  (reply to the message sender)
       "list"    (reply to the whole list)
       or enter the full e-mail address of who you want to receive replies
list reply_to setting [$SENDER] : <return>
    Enter a 50-char-max description of the list
list description [] : This is a test list
    Enter the desired reply_to configuration.  The choices are
       $SENDER  (reply to the message sender)
       "list"    (reply to the whole list)
       or enter the full e-mail address of who you want to receive replies
list owner [owner1@host.domain] : <return>
    Enter the full e-mail address(es) that can post a message to the list.
    If there is more than one person, you will get the chance to answer
    individually.  Just hit ‘return’ after the last address was entered.
list poster e-mail address (return to ‘break out’) [] : dude1@hostname.domain
list poster e-mail address (return to ‘break out’) [] : dude2@hostname.domain
list poster e-mail address (return to ‘break out’) [] : <return>
    Enter the type of list.  Choices are:
       permanent = the list stays forever
       temporary = the list ‘expires’ a few days after its use
```

```
enter list type (permanent|temporary) [permanent] : <return>

# mjdlm -add -list mytestlist
              MJDLM version 2.09
              ------------------
 found mytestlist      in  control staging         = create the list
....creating list "mytestlist"....
# mjdlm -show -list mytestlist
   name              = mytestlist
   owner             = owner1@hostname.domain
   description       = This is a test list
   reply_to          = $SENDER
   restrict_post     = dude1@hostname.domain dude2@hostname.domain
   num_sublists      = 0
   list_type         = permanent
   isa_sublist       = 0
   staged_recipients = mytestlist.permanent.19990708
   num_recipients    = 1234
```

Example sendmail aliases

```
# archived list => mytestlist
owner-mytestlist:    owner1@hostname.domain
mytestlist-owner:    owner1@hostname.domain
mytestlist-approval: owner1@hostname.domain
mytestlist:    "|/home/majordom/wrapper.dlm resend -l mytestlist \
                    mytestlist-list"
mytestlist-list:  :include:/home/majordom/mjdlm_data/listdir/mytestlist, \
                    mytestlist-archive
mytestlist-request: "|/home/majordom/wrapper.dlm majordomo \
                    -l mytestlist"
mytestlist-archive: "|/home/majordom/wrapper.dlm archive2.pl -f \
      /home/majordom/mjdlm_data/listdir/mytestlist.arch/mytestlist -m -a"
```