

# Tuning Sendmail for Large Mailing Lists

NLUUG  
Antwerp, Belgium

February 27, 1997

Rob Kolstad, Ph.D.  
<kolstad@bsd.com>



RBK - tunemal.00

Jul 9, 1997

Page 1

## Overview

- The problem
- Goals
- Initial state
- What we did (with graphs)
- Where we ended up

RBK - tunemal.01

Jul 9, 1997

Page 2

## The Problem

- Big mailing lists increasingly popular
  - News less useful
  - Push vs. pull
- Big mailing lists are big
  - 500-15,000 recipients
  - 1-100+ messages per day
- Message delivery slow and resource intensive

RBK - tunemal.02

Jul 9, 1997

Page 3

## Some Goals for Mailing Lists

- Fast delivery (low latency)
- Reasonable consumption of resources
- Existing tools
- Monitor results
- Easy to administer

RBK - tunemal.03

Jul 9, 1997

Page 4

### **Initial Situation**

- **inet-access@earth.com mailing list**
- **1,500 recipients**
- **40-110 messages per day**
- **Delivery times > 5 hours**
- **System load high, paging high**

RBK - tunenail.04

Jul 9, 1997

Page 5

### **Initial Data Gathering**

- **50,000 messages in mail queue**
- **> 100 sendmail processes**
- **Each process > 2.5 MB**
- **Load average high; CPU not fully used**
- **Disks medium busy**
- **Queued deliveries often catch up overnight**

RBK - tunenail.05

Jul 9, 1997

Page 6

### **Zereth Trial**

- **Tony Sanders sorted outgoing mail queue**
- **Gathered 'delivery stats' nightly**
- **Gave priority to those who responded quickly**
- **→ Great for those at front of queue**
  - **But still slow for those at end**

RBK - tunenail.06

Jul 9, 1997

Page 7

### **First Suggestion**

- **Split list into 75 lists of 20 each**
- **Completely overruled by sysadmins**
  - **Not enough RAM**
  - **Too many processes**
- **Kolstad unconvinced**

RBK - tunenail.07

Jul 9, 1997

Page 8

### Second Try, First Suggestion

- Split list into four lists of 375 each
- Sysadmins not pleased
- Kolstad wins these arguments; he's the president

RBK - tunenail.08

Jul 9, 1997

Page 9

### Results of First Suggestion

- Primitive analysis tools show 4x improvement in throughput
  - Looks like mail is delayed < 2 hours
  - Looks like delivery rate is increased
- Lots-o-processes
  - Lots of RAM use
  - Disks OK
- Hard to measure throughput cuz mail queue pieces only updated every ten deliveries
- So, we changed the update rate to update on every delivery
  - Improved monitoring
  - HUGE mistake

RBK - tunenail.09

Jul 9, 1997

Page 10

### Next Suggestion

- Trend toward more lists of fewer elements
- Construct more monitoring tools
  - Instantaneous rates
  - Summary pages
- Watch processes, disk I/O, network I/O

RBK - tunenail.10

Jul 9, 1997

Page 11

### Results

- Ever fewer elements → ever faster deliveries
- Single recipient sendmail processes never use much RAM (!)
- Disk I/O continuing to increase
- Network I/O very low
- Delivery times now below one hour (yay)
- Queue sizes now considerably reduced

RBK - tunenail.11

Jul 9, 1997

Page 12

### Head Scratching

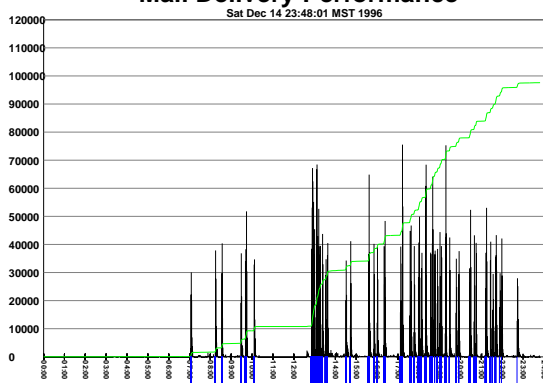
- RAM usage now low (!?)
- Where is/are the bottleneck(s)?
  - Disks are getting busy
  - CPU isn't that busy
  - Load average isn't getting worse
  - Network isn't the problem
  - Something else???
- What is the maximum throughput for mail delivery?

### Time to Build Monitoring Tools

- mailqq (slow! but easier way of seeing queues):

```
#!/bin/bash
mailq | perl -e '$junk = <STDIN>;
$junk = <STDIN>; $i = 0;
print "-##- ", $junk;
while (<STDIN>) {
    next if /^ /;
    if (/^\t/) { $i++; next unless
                eof(STDIN); }
    if ($line || eof(STDIN)) {
        printf("%5d %s\n", $i, $line);
        $sum += $i; $i = 0;
    }
    chop($line = $_);
}
printf "%5d TOTAL\n", $sum;'
```

### Batchstat Output Mail Delivery Performance



- Look at those spikes!

### Realstat Output During Slow Period

|         |       |    |      |
|---------|-------|----|------|
| 9:00:00 | 480/  | 2  |      |
| 9:00:15 | 960/  | 4  |      |
| 9:00:30 | 960/  | 4  | --   |
| 9:00:45 | 2160/ | 9  | *--  |
| 9:01:00 | 1920/ | 8  | *--  |
| 9:01:15 | 720/  | 3  | --   |
| 9:01:30 | 2400/ | 10 | **-- |
| 9:01:45 | 720/  | 3  | --   |
| 9:02:00 | 1440/ | 6  | *-   |
| 9:02:15 | 960/  | 4  | -    |
| 9:02:30 | 960/  | 4  | --   |
| 9:02:45 | 1440/ | 6  | *--  |

### Mailstat (Slow Day)

```
mailstat: Sat Feb 22 09:05:40 MST 1997
                failed      deliveries
MMM DD HH mhosts/recvpt mhosts/recvpt
==== == == =====
Feb 22 02      433/    612    1111/   1111
Feb 22 03      495/    696    1298/   1298
Feb 22 04      431/    615    1137/   1137
Feb 22 05      421/    610     810/    810
Feb 22 06      422/    606     717/    717
Feb 22 07      411/    587     931/    931
Feb 22 08      427/    616    1039/   1039
Feb 22 09       22/     22     105/    105
=====
Totals          3062/  4364    7148/   7148
```

RBK - tunemail.16

Jul 9, 1997

### Mailstat (Few Deliveries, Many failures)

```
mailstat: Sat Feb 22 09:03:49 MST 1997
                failed      deliveries
MMM DD HH mhosts/recvpt mhosts/recvpt
==== == == =====
Feb 22 03    18811/ 18811     522/    522
Feb 22 04    27065/ 27065     574/    574
Feb 22 05    29342/ 29346    1738/   1887
Feb 22 06    29973/ 29978         8/     8
Feb 22 07    26668/ 26675    1556/   1690
Feb 22 08    11768/ 11787    1347/   1464
Feb 22 09     1893/  1896     566/    606
=====
Totals    145520/145558    6311/   6751
```

RBK - tunemail.17

Jul 9, 1997

### What Is Max Speed?

- Wrote 'mailtest.c'
  - Opened mail port
  - Sent shortest possible message
  - Completed
- On PPro/200: ~30 ms → 120000 messages/hour
- Only bottlenecks were CPU and network

RBK - tunemail.18

Jul 9, 1997

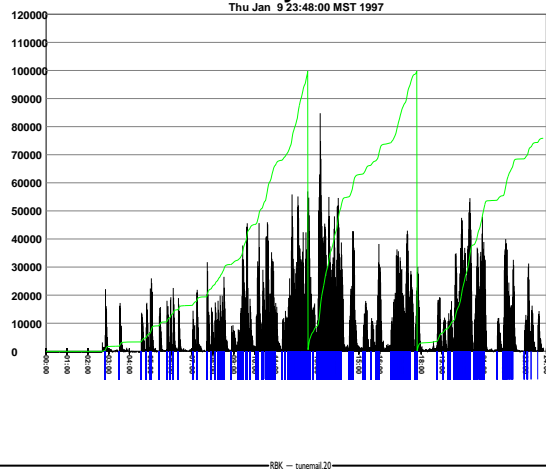
### Further Analysis

- Continued to increase parallelism
  - Went to 100 lists of 15-20 people each
  - Decreased delivery time
  - Machine very busy
  - Disks incredibly busy
- Stupid decision was to update stats for 'every delivery'
  - Great for real-time understanding of deliveries
  - /var/log/maillog vs. mailq
- Synchronous disk operations were destroying performance
- → So, went back to 'update every 10 deliveries'

RBK - tunemail.19

Jul 9, 1997

## Performance Increased Dramatically Mail Delivery Performance



Page 21

## Notes about Performance

- Note that peaks are very high
  - Even averaged, they are closing in on theoretical maximum
- Performance depends on other things, too
  - Number of hosts unavailable
  - Number of messages available to deliver (!)

RBK - tunemal.21 Jul 9, 1997

Page 22

## Next Analysis

- Why is mail queue so long?

```
[...]
nrecipients length date sender
3 BAA08598 1554 Sat Feb 22 01:02 <inet-access@earth.com>
3 BAA08677 1017 Sat Feb 22 01:29 <inet-access@earth.com>
3 FAA10201 1438 Sat Feb 22 05:24 <inet-access@earth.com>
3 FAA10208 1438 Sat Feb 22 05:24 <inet-access@earth.com>
3 HAA10369 1527 Sat Feb 22 07:46 <inet-access@earth.com>
3 IAA10524* 423 Sat Feb 22 08:52 <inet-access@earth.com>
4 HAA10371 1527 Sat Feb 22 07:46 <inet-access@earth.com>
4 HAA10383 1527 Sat Feb 22 07:46 <inet-access@earth.com>
4 IAA10544 423 Sat Feb 22 08:53 <inet-access@earth.com>
4 IAA10558 423 Sat Feb 22 08:53 <inet-access@earth.com>
5 IAA10541* 423 Sat Feb 22 08:53 <inet-access@earth.com>
```

- Each message has a few 'stragglers'

RBK - tunemal.22 Jul 9, 1997

Page 23

## What About Those Stragglers?

```
33 nathan@netrail.net
46 davek@melita.com
82 todd@acc.com
127 glennh@netstation.net
127 ispmail@zhi.dialup.access.net
127 jnuussbaum@americandata.net
127 kevinc@rrt.com
127 mp3@cyber-gate.com
127 nevin@shadowwave.com
127 tcosta@biznm.com
164 rdavis@masschaos.de.convex.com
200 whenpigsfly@worldsrv.net
388 cbrown@matnet.com
559 robert_thompson_at_usr-css1@robogate2.usr.com
593 berney.ortiz@mailserver.oig.state.pa.us
595 list.inet-access@optimum.net
```

- Small number of recipients never accepting mail!

RBK - tunemal.23 Jul 9, 1997

Page 24

### What Happens When Recipient Unavailable?

- **Sendmail tries to connect host**
  - Maybe succeeds (but host turns out to be slow or net is losing lots-o-packets)
  - Maybe not
- **Each step in the protocol from connection through completion has long time-out**
  - Like as much as 300 seconds
  - So that particular sendmail process idles for five minutes
  - → reduces throughput
- **Amazing fact: at any point in time, 1-3% of recipients are unavailable**
  - And these are ISPs!
  - But they can't control many kinds of outages

RBK - tuncel24

Jul 9, 1997

Page 25

### Next Step

- **Reduce timeouts for initial contact/mail transmission**
  - Configurable in sendmail
  - Reduced them 5x
- **Speeds up initial mail delivery**
  - Of course, some messages never delivered
  - So a second sendmail.cf file with slower timeouts was created; 'reaper' process runs 3x/hour

RBK - tuncel25

Jul 9, 1997

Page 26

### High Points of Modifications

- **Lots of parallelism (100 processes in parallel)**
- **Reduce impact of unavailable recipients**
- **Continued effects of stragglers**
  - mailq commands run really slowly
  - sendmail running entire queue is very slow
  - Needed to pay even more attention to stragglers to reduce queue search time

RBK - tuncel26

Jul 9, 1997

Page 27

### Reducing Queue Search Time

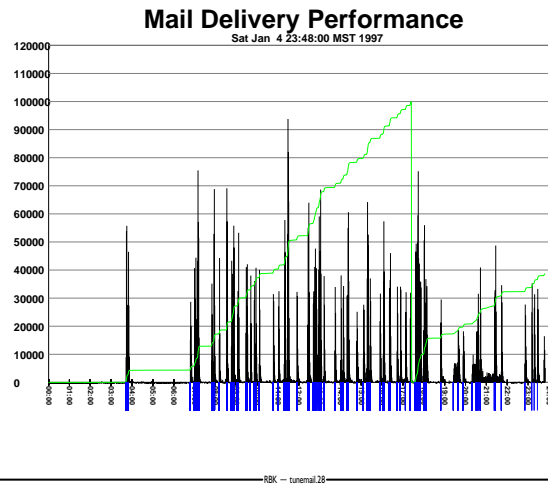
- **Can't remove people from list for one bounce – or even two days of bounces**
- **Created 10 more queues to run separately**
  - Jobs moved from queue to queue when older than specified amount
  - Ever more 'reaper' processes running those (presumably smaller) queues
- **Just never seemed to help :(**

RBK - tuncel27

Jul 9, 1997

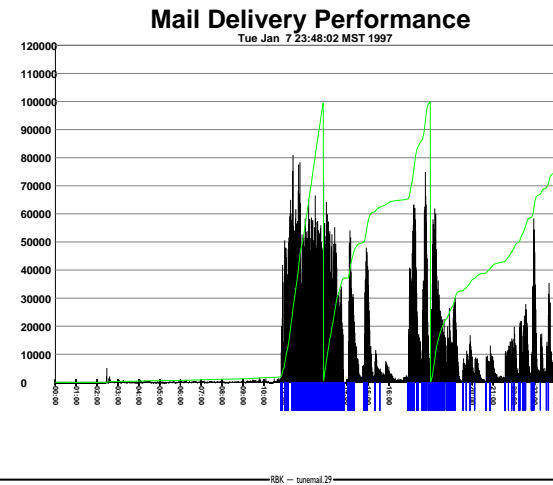
Page 28

## Mail Delivery Now FAST with High Spikes



Page 29

## High Load Performance Outstanding



Page 30

## Yet To Try

- Coalesce stragglers into single message/recipient-list pair
- Just delete notes older than, say, 24 hours from queue (they can always look at an archive)
- Rewrite sendmail for:
  - High constant number of transmission processes
  - Use of extended SMTP to send multiple messages once a machine is up
- But, to be fair, already fairly close to achieving highest possible bandwidth given speed of network connections
- Easy to use multiple CPUs to deliver more messages in parallel
  - And still not pushing T-1 speeds to limit yet (!)

RBK - tunemail.30 Jul 9, 1997

Page 31

## Conclusion

- Not difficult to reduce latency dramatically
- Kolstad has script to insert in /etc/aliases to break message into parts (kolstad@bsd.com for details - it's only 99% great)
- 95% mail delivery time was reduced from 5 hours to 3.5 minutes
- Unavailability of recipients still the biggest problem
- Probably about as good as can be done without redoing sendmail
  - But Russ Cox is writing a mail delivery agent
  - And has these results :)

RBK - tunemail.31 Jul 9, 1997

Page 32



