```perl
#!/usr/bin/perl -w

use strict;

$| = 1;
our(%msg, %delivery);

while (<>) {
  chomp;
  next unless s/.*qmail: //;
  s/(\d+\.\d+)\s+//;
  my $time = $1;
  if (/^new msg (\d+)/) {
    if (exists $msg{$1}) {
      warn "Duplicate message number $1";
      next;
    }
    $msg{$1}{arrived} = $time;
    $msg{$1}{msg} = $1;
    $msg{$1}{bounced} = '-';
  } elsif (/^info msg (\d+): bytes (\d+) from <([^>]*)>/) {
    $msg{$1}{length} = $2;
    $msg{$1}{from} = $3 || "-";
  } elsif (/^starting delivery (\d+): msg (\d+) to (local|remote) (.*)/) {
    $delivery{$1} = {lr => substr($3, 0, 1),
                     to => $4,
                     started => $time};
    $msg{$2}{del}{$1} = $delivery{$1};
  } elsif (/^delivery (\d+): (success|deferral|failure): (.*)/) {
    unless (exists $delivery{$1}) {
      warn "Delivery $1 unknown";
      next;
    }
    $delivery{$1}{result} = $2;
    $delivery{$1}{report} = $3;
    $delivery{$1}{finished} = $time;
  } elsif (/^end msg (\d+)/) {
    $msg{$1}{finished} = $time;
    report($msg{$1});
    for my $del (keys %{$msg{$1}{del}}) {
      delete $delivery{$del};
    }
    delete $msg{$1};
  } elsif (/^status:/) {
    next;
  } elsif (/^bounce msg (\d+)/) {
    $msg{$1}{bounced} = 'B';
  } else {
#   warn "Unmatched line: $_\n";
  }
}

sub report {
  my $msg = shift;
  # Fill me in
}
```